AD-A258 899

①

DTIC
S ELECTE D
JAN 8 1993
C

KALMAN FILTER TRACKING OF A BALLISTIC MISSILE USING FORWARD LOOKING INFRARED MEASUREMENTS AND DOPPLER RETURNS

THESIS

Michael L.G. Ching
Captain, USAF

AFIT/GE/ENG/92D-09

DISTRIBUTION STATEMENT A
Approved for public release
Distribution Unlimited

AFIT/GE/ENG/92-D-09

KALMAN FILTER TRACKING OF A BALLISTIC MISSILE USING FORWARD LOOKING INFRARED MEASUREMENTS AND DOPPLER RETURNS

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Electrical Engineering

Michael L.G. Ching, B.S.E.E.
Captain, USAF

Accession For
NTIS GRA&I ☑
DTIC TAB ☐
Unannounced ☐
Justification ____
By ____
Distribution/
Availability Codes
Dist | Avail and/or Special
A-1

93-00063

DTIC QUALITY INSPECTED 5

Approved for public release; distribution unlimited

93 1 04 164

Acknowledgement

This thesis, sponsored by the Phillips Laboratory, at Kirtland AFB, New Mexico, continues the development of a Kalman filter that uses a forward-looking infrared sensor and low-energy laser measurements to track airborne targets. It is the most recent in a lengthy research program existing since 1978 and specifically concentrated on combining the measurements from both sensors into a single filter along with modeling the plume "pogo" affect I feel honored to have been afforded the opportunity to contribute to this research and deeply admire the achievements that have been made during the project. This thesis represents the culmination of 18 months of painful yet challenging and ultimately rewarding graduate studies. The learning experience gained throughout my endeavors would not have been complete without the help of several people.

First and foremost, I wold like to thank my friends and comrades for helping me through the tough times. There are very few places where I've had the privilege of being associated with such extreme professionalism, competence, and enthusiasm in a collection of people. The AFIT pressure cooker can be difficult to bear and the support of my peers went a long way toward helping me "blow off steam" and handle the stress. The exchange of both friendship and technical knowledge that took place during our "AFIT experience" will not be forgotten.

I would also like to acknowledge my predecessors, whose vast intellect was clearly evident in their work. The legacy left behind in their theses was invaluable to helping me understand and conquer the objectives of my research.

Finally, I would like to thank Dr. Peter Maybeck. I've come to understand why he has been acknowledged in virtually every previous thesis for his infinite patience and understanding and seemingly boundless wisdom. He had the uncanny ability to help me expand my mental horizons beyond what I thought possible. It was my pleasure and good fortune to have Dr. Maybeck as an advisor. His guidance and insight into my problems were the key to my success.

To everyone who made AFIT such a memorable experience, THANKS!!!

Michael L.G. Ching

Table of Contents

Acknowledgement ii

List of Figures xi

List of Tables xvi

Abstract xvii

I. Introduction 1-1

- 1.1 Background 1-4
- 1.2 Summary of Previous Research 1-6
- 1.3 Thesis Objectives 1-25
  - 1.3.1 Observability Gramian Analysis 1-26
  - 1.3.2 Sensitivity Analysis of Single Filters 1-29
  - 1.3.3 Incorporation of Laser Measurements into Elemental Filters 1-31
  - 1.3.4 Multiple Model Adaptive Filter Design and Analysis 1-32
- 1.4 Thesis Overview 1-33

II. Filtering Techniques and Reference Coordinate Systems 2-1

- 2.1 Introduction 2-1
- 2.2 Kalman Filter Theory 2-2

2.2.1 Introduction 2-2

2.2.2 Linear Kalman Filter 2-3

2.2.3 Multiple Model Adaptive Algorithm 2-7

2.3 Simulation Space 2-11

2.3.1 Coordinate Frames 2-12

2.3.1.1 Inertial Reference Frame 2-12

2.3.1.2 Target Plume Reference Frame 2-12

2.3.1.3 α - β - r Reference Frame 2-13

2.3.1.4 α - β (FLIR Image) Plane 2-14

2.3.1.5 Absolute α-β-r Reference Frame 2-14

2.3.1.6 Trans-FLIR Plane 2-15

2.3.1.7 ALT/ACT Plane 2-15

2.3.2 FLIR Image Plane 2-15

2.3.2.1 FLIR Field-Of-View 2-15

2.3.2.2 Target Models on the FLIR Plane 2-17

2.3.2.3 Target Plume Model on the FLIR Plane 2-18

2.3.2.4 Target Plume Projection onto the FLIR Plane 2-20

2.3.2.5 Target Plume Velocity Projection onto the FLIR Plane . . 2-22

2.3.3 FLIR Sensor Pointing Controller 2-25

2.3.4 Target Scenarios 2-26

2.4 Summary 2-27

III. Truth Model 3-1

3.1 Introduction 3-1

3.2 Truth Dynamics Model 3-2

3.2.1 Target Dynamics State Description 3-7

3.2.1.1 AFIT Discrete-Time Target Dynamics 3-9

3.2.1.2 MSOFE Continuous-Time Target Dynamics 3-10

3.2.2 Atmospheric Jitter Model 3-11

3.2.3 Bending/Vibration Model 3-14

3.2.4 Plume Pogo Model 3-17

3.3 Measurement Models 3-20

3.3.1 FLIR Model 3-21

3.3.2 Low-Energy Laser Speckle Reflection Model 3-25

3.3.2.1 The Hardbody Reflectivity Model 3-25

3.3.2.2 Plume Reflectance Model 3-32

3.3.3 The Doppler Measurement Model 3-35

3.3.3.1 The Doppler Effect 3-36

3.3.3.2 Hardbody Doppler Return 3-39

3.3.3.3 Plume Doppler Return 3-40

3.3.3.4 Doppler Measurement Noises 3-44

3.4 Truth Model Parameters 3-48

3.4.1 Target Trajectory Initial Conditions 3-48

3.4.2 Target Model, Dimensions, and Orientation 3-48

3.4.3 Intensity Functions 3-49

3.4.4 Atmospheric Jitter 3-49

3.4.5 Bending/Vibration 3-49

3.4.6 Plume Pogo Characteristics 3-49

3.4.7 Spatially Correlated Background Noise 3-49

3.4.8 Low-Energy Laser Speckle Return Measurement Dimensions 3-50

3.4.9 Plume Reflectance Model 3-50

3.4.10 Low-Energy Doppler Return Measurement Dimensions 3-50

3.4.11 Hardbody Reflectivity Measurement Model 3-50

3.5 Summary 3-51

IV. Filter Models 4-1

4.1 Introduction 4-1

4.2 Dynamics Models 4-2

4.2.1 Target Dynamics Model 4-6

4.2.2 Atmospheric Disturbance Model 4-7

4.2.3 Pogo Dynamics Model 4-9

4.2.4 Centroid Equilibrium Point/Center-of-Mass Offset Model 4-10

4.3 Measurement Models 4-11

4.3.1 FLIR Measurement Model 4-11

4.3.1.1 Enhanced Correlator Algorithm 4-12

4.3.1.2 Template Generation 4-14

4.3.1.3 "Pseudo-Measurements" 4-16

4.3.2 Doppler Measurement Model 4-22

4.3.3 Filter Parameters 4-26

4.3.3.1 Initial Conditions 4-26
4.3.3.2 Tuning Values 4-27
4.4 Summary 4-28

V. Observability Gramian and MSOFE Analysis 5-1
5.1 Introduction 5-1
5.2 Stochastic Observability Tests 5-2
5.2.1 Background Review 5-2
5.2.2 The Stochastic Observability Test 5-4
5.2.3 Observability Test Results and Conclusions 5-5
5.3 MSOFE Monte Carlo Analysis 5-7
5.3.1 MSOFE Filter Configurations 5-9
5.3.2 Monte Carlo Analysis Results 5-12
5.3.2.1 Linear Filter (Constant θ_f) vs Nonlinear Filter 5-12
5.3.2.2 Position Accuracy 5-13
5.3.2.3 Atmospheric Jitter Estimation 5-14
5.3.2.4 Plume Pogo Estimation 5-15
5.3.2.5 Plume Pogo/Atmospheric Jitter Interaction 5-15
5.3.2.6 Sensitivity to Different Plume Pogo Conditions 5-16
5.4 Summary 5-18

VI. AFIT Elemental Filter Analysis Procedures and Results 6-1
6.1 Introduction 6-1

6.2 Performance Analysis Results 6-3

6.2.1 Position Estimation Accuracy 6-4

6.2.2 Velocity Estimation Accuracy 6-4

6.2.3 Atmospheric Jitter Estimation Accuracy 6-5

6.2.4 Plume Pogo Position and Velocity Estimation Accuracy 6-5

6.2.5 Offset Estimation Accuracy 6-7

6.2.6 Plume Intensity Centroid Estimation Accuracy 6-7

6.3 Potential Problems and Proposed Solutions 6-8

6.3.1 Specific Problems Revealed by Analysis 6-8

6.3.2 Miscellaneous Issues 6-11

6.4 Summary 6-13

VII. Conclusions and Recommendations 7-1

7.1 Introduction 7-1

7.2 Conclusions 7-1

7.2.1 Stochastic Observability Tests 7-1

7.2.2 MSOFE Monte Carlo Analysis Results 7-2

7.2.3 AFIT Elemental Filter Performance 7-3

7.3 Recommendations 7-5

7.3.1 Stochastic Observability 7-6

7.3.2 MSOFE Monte Carlo Analysis 7-6

7.3.3 AFIT Elemental Filter Research 7-7

7.3.4 General Software Upgrades 7-8

Appendix A: Data Processing Statistics Method A-1

Appendix B: Examples of Performance Plots B-1

Appendix C: MSOFE Analysis of Filter Models Using Linear Update C-1

Appendix D: MSOFE Analysis Using Varying Pogo Conditions D-1

Appendix E: AFIT Elemental Filter Performance Plots E-1

Bibliography Bib-1

Vita VITA-1

List of Figures

Figure 1.1 Ballistic Missile Tracking Scenario 1-2

Figure 1.2 Multiple Model Adaptive Filter 1-10

Figure 1.3 Data Processing Scheme using FFT and IFFT 1-11

Figure 1.4 Linear Kalman Filter/Enhanced Correlator Algorithm 1-13

Figure 1.5 Discrete Implementation of Cross-Sectional Reflectivity Function 1-22

Figure 1.6 Pogo Phenomenon 1-25

Figure 2.1 Three Primary Coordinate Frames in Simulation Space 2-13

Figure 2.2 FLIR Plane, Trans-FLIR Plane, and ALT-ACT Plane 2-16

Figure 2.3 Target Plume Image in 8 x 8 FLIR Field-of-View (FOV) 2-17

Figure 2.4 Spatial Relationship of Target Plume Gaussian Intensity Functions 2-19

Figure 2.5 Target Plume Intensity Centroid Projection Geometry 2-21

Figure 2.6 Intensity Centroid Dispersion Axes in FLIR Plane 2-22

Figure 2.7 Inertial Velocity FLIR Plane Projection Geometry 2-24

Figure 3.1 Plume Intensity Function Position on FLIR Image Plane 3-3

Figure 3.2 Target Centroid Image on α-β Plane with "Pseudo" Angles 3-8

Figure 3.3 Composite Plume Intensity Function on FLIR Plane 3-22

Figure 3.4 3-d Hardbody Reflectivity Model Relative to FLIR Image Plane 3-27

Figure 3.5 Discrete Implementation of Cross-Sectional Reflectivity Function [9] 3-28

Figure 3.6 Projection Geometry onto FLIR Image Plane 3-30

Figure 3.7 Ideal Low-Energy Laser Scan 3-31

Figure 3.8 Sweep Technique of Laser Scan 3-32

Figure 3.9 Biased Offset Measurement Caused by Plume Reflectance 3-34

Figure 3.10 Spectra of Received Signals [43] 3-38

Figure 3.11 Pulsed Signal Frequency Spectrum [44] 3-40

Figure 3.12 Spectra of Coherent and Non-Coherent Pulsed Signals [44] 3-41

Figure 3.13 Spectra of Plume and Hardbody-Induced Doppler Returns 3-42

Figure 3.14 Detection Characteristics [43] 3-47

Figure 4.1 Linear Kalman Filter/Enhanced Correlator Algorithm 4-12

Figure 4.2 Filter Estimate of Offset Distance (Ideal Conditions) 4-23

Figure B.1 Example of State Plot and Error Statistics Plot B-3

Figure C.1 Simulation #2, θ = 60°, X-Position C-2

Figure C.2 Simulation #2, θ = 60°, Y-Position C-3

Figure C.3 Simulation #2, θ = 60°, X-Velocity C-4

Figure C.4 Simulation #2, θ = 60°, Y-Velocity C-5

Figure C.5 Simulation #4, θ = 60°, X-Position C-6

Figure C.6 Simulation #4, θ = 60°, Y-Position C-7

Figure C.7 Simulation #4, θ = 60°, X-Velocity C-8

Figure C.8 Simulation #4, θ = 60°, Y-Velocity C-9

Figure C.9 Simulation #4, θ = 60°, X-Atmospheric Jitter C-10

Figure C.10 Simulation #2, θ = 60°, Y-Atmospheric Jitter C-11

Figure C.11 Simulation #6, θ = 60°, X-Position C-12

Figure C.12 Simulation #6, θ = 60°, Y-Position C-13

Figure C.13 Simulation #6, θ = 60°, X-Velocity C-14

Figure C.14 Simulation #6, θ = 60°, Y-Velocity C-15

Figure C.15 Simulation #6, $\theta = 60^\circ$, Pogo Offset C-16

Figure C.16 Simulation #7, $\theta = 60^\circ$, X-Position C-17

Figure C.17 Simulation #7, $\theta = 60^\circ$, Y-Position C-18

Figure C.18 Simulation #7, $\theta = 60^\circ$, X-Velocity C-19

Figure C.19 Simulation #7, $\theta = 60^\circ$, Y-Velocity C-20

Figure C.20 Simulation #8, $\theta = 60^\circ$, X-Position C-21

Figure C.21 Simulation #8, $\theta = 60^\circ$, Y-Position C-22

Figure C.22 Simulation #8, $\theta = 60^\circ$, X-Velocity C-23

Figure C.23 Simulation #8, $\theta = 60^\circ$, Y-Velocity C-24

Figure C.24 Simulation #8, $\theta = 60^\circ$, Pogo Offset C-25

Figure C.25 Simulation #9, $\theta = 60^\circ$, X-Position C-26

Figure C.26 Simulation #9, $\theta = 60^\circ$, Y-Position C-27

Figure C.27 Simulation #9, $\theta = 60^\circ$, X-Velocity C-28

Figure C.28 Simulation #9, $\theta = 60^\circ$, Y-Velocity C-29

Figure C.29 Simulation #9, $\theta = 60^\circ$, X-Atmospheric Jitter C-30

Figure C.30 Simulation #9, $\theta = 60^\circ$, Y-Atmospheric Jitter C-31

Figure C.31 Simulation #10, $\theta = 60^\circ$, X-Position C-32

Figure C.32 Simulation #10, $\theta = 60^\circ$, Y-Position C-33

Figure C.33 Simulation #10, $\theta = 60^\circ$, X-Velocity C-34

Figure C.34 Simulation #10, $\theta = 60^\circ$, Y-Velocity C-35

Figure C.35 Simulation #10, $\theta = 60^\circ$, X-Atmospheric Jitter C-36

Figure C.36 Simulation #10, $\theta = 60^\circ$, Y-Atmospheric Jitter C-37

Figure C.37 Simulation #10, $\theta = 60^\circ$, Pogo Offset C-38

Figure D.1 Pogo Condition #1, Simulation #10, General Angle, X-Position D-2

Figure D.2 Pogo Condition #1, Simulation #10, General Angle, Y-Position D-3

Figure D.3 Pogo Condition #1, Simulation #10, General Angle, X-Atmospheric Jitter .. D-4

Figure D.4 Pogo Condition #1, Simulation #10, General Angle, Y-Atmospheric Jitter .. D-5

Figure D.5 Pogo Condition #1, Simulation #10, General Angle, Pogo Offset D-6

Figure D.6 Pogo Condition #2, Simulation #10, General Angle, X-Position D-7

Figure D.7 Pogo Condition #2, Simulation #10, General Angle, Y-Position D-8

Figure D.8 Pogo Condition #2, Simulation #10, General Angle, X-Atmospheric Jitter .. D-9

Figure D.9 Pogo Condition #2, Simulation #10, General Angle, Y-Atmospheric Jitter . D-10

Figure D.10 Pogo Condition #2, Simulation #10, General Angle, Pogo Offset D-11

Figure D.11 Pogo Condition #3, Simulation #10, General Angle, X-Position D-12

Figure D.12 Pogo Condition #3, Simulation #10, General Angle, Y-Position D-13

Figure D.13 Pogo Condition #3, Simulation #10, General Angle, [illegible]-Atmospheric Jitter . D-14

Figure D.14 Pogo Condition #3, Simulation #10, General Angle, Y-Atmospheric Jitter . D-15

Figure D.15 Pogo Condition #3, Simulation #10, General Angle, Pogo Offset D-16

Figure D.16 Pogo Condition #4, Simulation #10, General Angle, X-Position D-17

Figure D.17 Pogo Condition #4, Simulation #10, General Angle, Y-Position D-18

Figure D.18 Pogo Condition #4, Simulation #10, General Angle, X-Atmospheric Jitter . D-29

Figure D.19 Pogo Condition #4, Simulation #10, General Angle, Y-Atmospheric Jitter . D-20

Figure D.20 Pogo Condition #4, Simulation #10, General Angle, Pogo Offset D-21

Figure D.21 Pogo Condition #5, Simulation #10, General Angle, X-Position D-22

Figure D.22 Pogo Condition #5, Simulation #10, General Angle, Y-Position D-23

Figure D.23 Pogo Condition #5, Simulation #10, General Angle, X-Atmospheric Jitter . D-24

Figure D.24 Pogo Condition #5, Simulation #10, General Angle, Y-Atmospheric Jitter . D-25

Figure D.25 Pogo Condition #5, Simulation #10, General Angle, Pogo Offset D-26

Figure D.26 Pogo Condition #6, Simulation #10, General Angle, X-Position D-27

Figure D.27 Pogo Condition #6, Simulation #10, General Angle, Y-Position D-28

Figure D.28 Pogo Condition #6, Simulation #10, General Angle, X-Atmospheric Jitter . D-29

Figure D.29 Pogo Condition #6, Simulation #10, General Angle, Y-Atmospheric Jitter . D-30

Figure D.30 Pogo Condition #6, Simulation #10, General Angle, Pogo Offset D-31

Figure E.1 X/Y Position Error Statistics E-2

Figure E.2 X/Y Velocity Error Statistics E-3

Figure E.3 X Atmospheric Jitter E-4

Figure E.4 Y Atmospheric Jitter E-5

Figure E.5 Pogo Offset E-6

Figure E.6 Pogo Velocity E-7

Figure E.7 Center-of-Mass to Equilibrium Point Offset E-8

Figure E.8 Plume Centroid Error Statistics E-9

List of Tables

Table 1.1 Kalman Filter and Correlation Tracker Statistics Comparison 1-8

Table 1.2 Monte Carlo Analysis Combinations 1-29

Table 5.1 Filter States for Stochastic Observability Tests 5-5

Table 5.2 Filter/Truth Model Combinations Analyzed with MSOFE 5-9

Table 5.3 Pogo Conditions for Nonlinear Filter Analysis 5-11

Table 5.4 Pogo Error Statistics for Different Pogo Conditions 5-17

Table 6.1 Time-Averaged Error Statistics for Elemental Filter (Nominal Pogo Conditions) 6-3

Abstract

The Air Force Institute of Technology (AFIT) has been involved in developing Kalman filter trackers for airborne targets for the last 14 years. The goal of this particular thesis was to track a ballistic missile in the boost phase at ranges up to 2000 km, in order to control a high energy laser weapon designed to destroy the target. The filter developed combined an existing "FLIR" filter, which estimated location of the plume intensity centroid based on measurements from a forward looking infrared sensor (FLIR), and an existing "center-of-mass" filter, which estimated the offset between the plume and missile center-of-mass based on measurements from low-energy laser reflections. In addition, the new filter modeled the oscillation of the rocket plume with respect to the missile hardbody, known as the "pogo" affect, in the hopes of improving overall tracking performance. Filter performance is analyzed through use of Monte Carlo simulation software developed at AFIT.

This thesis also performed observability tests on various filter configurations in order to gain insight into observability problems identified during earlier research. Observability of states is measured through the use of both stochastic observability testing and Monte Carlo analysis of the filter models using the Multimode Simulation for Optimal Filter Evaluation (MSOFE) software.

KALMAN FILTER TRACKING OF A BALLISTIC MISSILE USING FORWARD-LOOKING INFRARED AND DOPPLER RETURN MEASUREMENTS

I. Introduction

The radical changes in the geopolitical complexion of the world have triggered corresponding drastic adjustments to the defensive posture of the Unites States military complex. Despite major cutbacks, the US Government continues its commitment to develop a national ballistic missile defense system under the Strategic Defense Initiative (SDI). One of those technologically challenging programs within SDI is the research to design a space-based contingent that can destroy ballistic missiles immediately after launch while they are still in the boost phase.

There are numerous technologies that must be employed to solve the problem of acquiring, tracking, and pointing a high energy laser (HEL) weapon at a ballistic missile. The major technologies exploited in the solution to this problem are the Kalman filter, the laser, and the forward looking infrared sensor (FLIR). Although all three technologies matured independently, they must be combined in a single system to solve this problem.

One of the SDI systems currently in development at the Phillips Laboratory, Kirtland Air Force Base, New Mexico, is a space-based system that contains a defensive weapon, the HEL, and all associated sensors and hardware necessary for tracking and pointing the HEL. The main

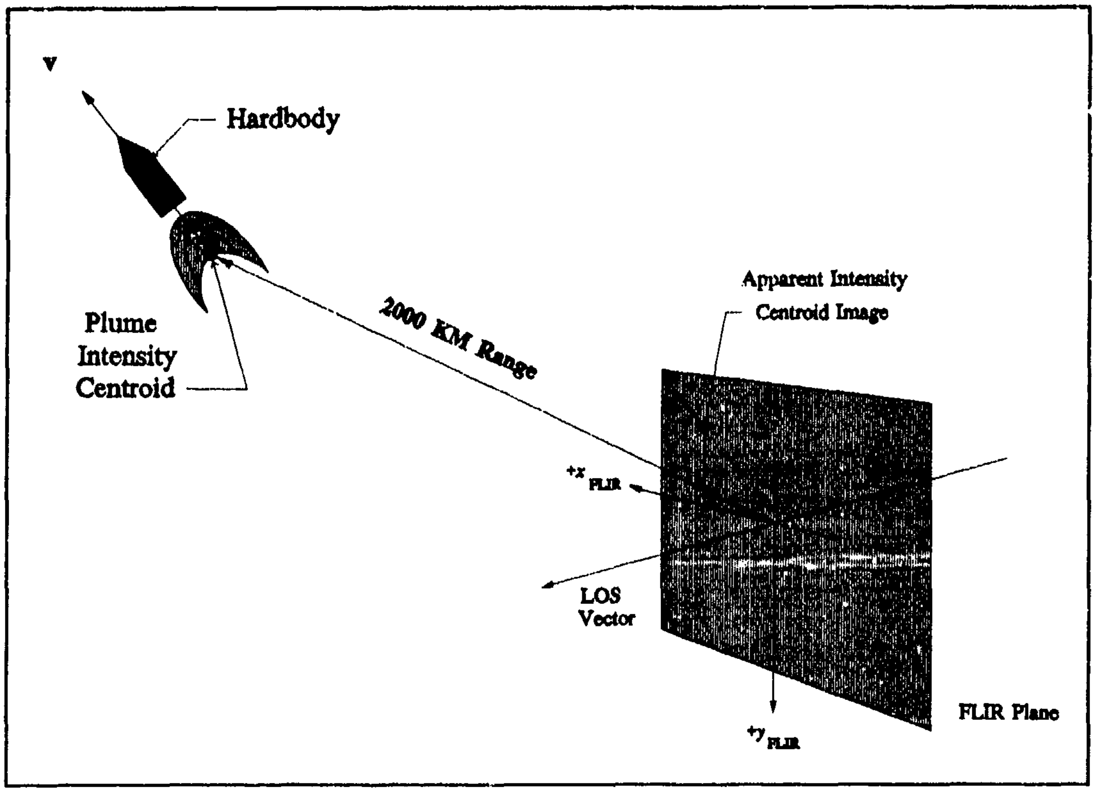


Figure 1.1 Ballistic Missile Tracking Scenario

objective, shown pictorially in Figure 1.1, is to acquire a launched missile, track the missile precisely, and point and fire the laser. Figure 1.1 depicts a simplified relationship between the FLIR and the target. The FLIR plane shown is associated with tracker. The tracker centers the line-of-sight (LOS) vector of the FLIR plane on the target. It is eventually desired to point the LOS vector directly at the missile hardbody, since this is a shared optics device such that the laser will fire along that LOS. Since tracking and firing must take place rapidly, at extremely long distances, and through the atmosphere, this objective is quite challenging to achieve. A number of major difficulties that must be overcome in this tracking problem are considered and implemented in this thesis research. Atmospheric disturbances cause the image of the rocket plume image to jitter on the FLIR plane. The atmosphere will also attenuate and jitter both the low energy laser (LEL) sensor (to be discussed subsequently) and the HEL, but to different extents. Bending of the space platform itself will also cause errors in pointing accuracy. The original tracker incorporated measurements from only the FLIR; however, it obtains information only on the high intensity infrared image of the plume. It is essential to locate the missile hardbody versus its plume centroid for firing the HEL. A phenomenon known as pogo causes the plume to oscillate with respect to the missile and makes estimation of the missile center-of-mass location difficult (if not impossible) based on FLIR data alone. For this reason, measurements from the LEL were incorporated to gain direct measurement information on the hardbody. Once FLIR measurements allow estimated motion of the plume center-of-intensity position and target velocity, the LEL is swept along the estimated velocity vector (as seen in the FLIR image plane), starting from the estimated center-of-intensity, in anticipation of intercepting the missile hardbody. Using reflected information from the LEL, the locations of the plume/hardbody interface and the hardbody-nose/space-background interface would yield an estimate of the hardbody center-of-mass location. It was determined that use of speckle reflectance information from the LEL did not allow clear distinction of the hardbody/plume interface and therefore resulted in biased calculations of the hardbody center-of-mass. To alleviate this difficulty, a method to utilize the doppler shift frequencies in the return was implemented to define the interface clearly.

These anomalies must be considered in design of a tracker and compensated, for such that the HEL beam can be maintained on a spot on the missile at a 2000 Km range during the firing sequence. To complicate matters further, the laser is not capable of destroying the missile instantaneously (like a kinetic kill weapon) but must be maintained within a small area on the target over a relatively long period of time [4]. This research specifically addresses the difficulties with optimally combining measurements from the FLIR and LEL in order to locate the missile center-of-mass in the presence of plume pogo.

1.1 Background

The Phillips Laboratory (formerly the Air Force Weapons Laboratory or AFWL), at Kirtland Air Force Base, New Mexico, has been sponsoring research for the past 14 years at the Air Force Institute of Technology (AFIT) in the use of directed energy weapons to destroy airborne targets [41]. This section provides a general overview of the sensor/tracker system in development by this research.

The primary sensor being used to track the targets is a 300 x 500 picture element (pixel) array FLIR sensor. The FLIR is a passive sensor which detects the infrared radiation emitted by the missile plume. Each pixel in the array discerns infrared energy through an angle of 15 microradians in two orthogonal directions (azimuth and elevation) [40]. A smaller 8 x 8 pixel array is utilized as the tracking window, which will be referred to as the field of view (FOV) in this thesis [14,40]. The tracker has been found to be accurate enough to maintain the target within the smaller FOV, and the smaller number of pixels allows for less computational loading and time lag than required for a larger number of pixels.

Excitations from detectors in the 8 x 8 FOV are fed to an enhanced image correlation algorithm (presented later in Figure 1.4). This algorithm compares the current raw FLIR data frame with a template which estimates the plume's intensity function [41]. The correlation then determines the optimal offsets in two orthogonal directions (defining the FLIR plane) that yield maximum correlation with the template. These offsets are the "pseudo-measurements" for a linear

Kalman filter. The linear Kalman filter updates position estimates of the plume centroid location based on the pseudo-measurements and propagates an estimate of the offset (from the current center of the FOV) of the centroid at the next update frame. A controller applies the appropriate adjustments to the FLIR plane in order to keep the centroid in the center of the FOV, i.e., to point the LOS vector of Figure 1.1 at the predicted location of the target one sample period later.

The previous three theses [8,9,14] incorporated use of a second sensor, a low energy laser, to obtain measurements of the location of the hardbody itself. The LEL is swept along the estimated target velocity vector in the FLIR image plane, relative to the centroid estimate, in order to obtain the center-of-mass location. Prior research incorporated the LEL measurements into the tracking algorithm using two different methods. Either a single linear Kalman filter (with decoupled models for incorporating the two types of measurements) accepted measurements from both the FLIR and the LEL, or two independent Kalman filters utilized the measurements separately. In either case, two independent estimates of intensity centroid location and the offset of the center-of-mass from that location were combined outside the filter to determine where to point the HEL.

This research incorporates both types of measurements into a single Kalman filter that optimally combines the measurements to estimate the location of the missile center-of-mass directly. Unlike previous research, there will be no purposeful attempt to decouple the two types of measurements, but to process them with a full accounting for physical interrelationships.

1.2 Summary of Previous Research

The Air Force Institute of Technology has been engaged in supporting the Phillips laboratory in the development of ballistic missile trackers since 1978 [31]. Sixteen theses and a number of other documents report the ongoing developments at AFIT. Each thesis contains a synopsis of previous work. This section provides a general overview of the previous ballistic missile tracker research completed at AFIT. The information included in this section is essentially a replication of the version from Herrera's thesis [14] with minor alterations and comments added to include his own contributions.

Research in this area was initiated by Mercier [31] in 1978, who compared extended Kalman filter (EKF) performance to that of the AFWL correlation tracker under identical conditions. An eight-state truth model was developed for simulation purposes, consisting of two target position states and six atmospheric jitter states. The position states defined the target location in each of two FLIR plane coordinate directions (azimuth and elevation), by accurately portraying target trajectories in three-dimensional space and projecting onto the FLIR plane. The atmospheric jitter was modeled by a third order shaping filter driven by white noise for each FLIR plane axis, as provided by The Analytic Sciences Corporation (TASC) [21]; three states defined the atmospheric distortion in each of the two FLIR plane coordinate directions. The Kalman filter dynamics model consisted of four states: two states representing target position, and two representing the atmospheric jitter (based on reduced order models, versus the six states of the truth model). In both the truth model and filter dynamics model, the position states and atmospheric jitter states were defined in each of the two FLIR plane coordinate directions. In the filter, the position and jitter states were each modeled as a first-order, zero-mean, Gauss-Markov process. The FLIR provided sampled data measurements to the filter at a 30 Hertz (Hz) rate. The FLIR measurement noises corrupting each pixel output due to background clutter effects and internal FLIR noises were modeled in the filter as both temporally and spatially uncorrelated. The target was considered as a point source of light (i.e., a long range target) having benign dynamics. The corresponding Airy disc on the FLIR image plane was modeled as a bivariate Gaussian distribution with circular equal intensity contours. The conventional correlation tracker and the extended Kalman filter were compared across three different signal-to-noise ratios (SNR), using a ten-run Monte Carlo analysis to obtain the tracker error statistics. The results of the comparison are shown in Table 1.1 for a Gaussian intensity function dispersion, σ_g, equal to one pixel. (For a Gaussian intensity function dispersion equal to one pixel, most of the useful information is contained in an area of about five pixels square.)

While the correlation tracker showed dramatic performance degradation as the SNR was decreased, the Kalman filter showed only a minor change in its performance at the lowest SNR tested. The extended Kalman filter was shown to be superior to the correlation tracker by an order of magnitude in the root mean square (rms) tracking error, provided the models incorporated into the filter were a valid depiction of the tracking scenario. This success motivated a follow-on thesis to improve filter modeling and thereby to enhance the performance.

The research accomplished by Harnly and Jenson [13,25] investigated modeling improvements in the filter and tested more dynamic target simulations. A comparison was made between a new six-state filter and a new eight-state filter. The six-state filter dynamics target model included the four previous states as well as two velocity states in the FLIR plane coordinates (azimuth and elevation); the dynamics model of the eight-state filter included two

Table 1.1 Kalman Filter and Correlation Tracker Statistics Comparison

Signal-to Noise Ratio	Correlation Tracker		Extended Kalman Filter	
	Mean Error	1σ	Mean Error	1σ
20	7.0	8.0	0.0	0.2
10	8.0	10.0	0.0	0.2
1	15.0	30.0	0.0	0.8

acceleration states in the FLIR coordinates as well. The acceleration was modeled as Brownian motion (BM) ($\dot{\mathbf{a}} = \mathbf{w}$, where $\mathbf{w}$ is a zero-mean white Gaussian noise). The filter was also designed to perform residual monitoring, which allowed the filter to react adaptively, and maintain track, by quickly increasing the values in the filter-computed state covariance matrix $\mathbf{P}$, which in turn increased the filter gain $\mathbf{K}$. A recommendation was also made to examine increasing the FOV during target jinking maneuvers to avoid losing lock. The constant-intensity contours of the target were modeled as elliptical patterns as opposed to the earlier circular equal-intensity contours in order to simulate closer range targets. The major axis of the target FLIR image was aligned with the estimated velocity vector. A number of different target trajectories were tested against the six-state and eight-state filters, and while the six-state filter performed well during moderate jinking maneuvers, the eight-state filter performed better while tracking high-g target maneuvers.

Other approaches to modeling the dynamics of the target in the filter were considered by Flynn [11]. He used a Brownian motion (BM) acceleration target dynamics model [13] and a constant turn-rate (CTR) dynamics model. The CTR model portrayed the target behavior by modeling the acceleration as that associated with CTR dynamics. Concatenating such constant turn-rate segments together provides an accurate portrayal of manned target evasive maneuver trajectories. Additionally, a Bayesian multiple model adaptive filter (MMAF) was developed using the BM dynamics model. A MMAF (Figure 1.2) consists of a bank of K independent Kalman filters, each of which is tuned to a specified target dynamics characteristic or parameter (a_1, a_2,...a_K in Figure 1.2). The time histories of the residuals ($r_K(t_i)$ in Figure 1.2) of these K Kalman filters are processed to compute the conditional probability ($p_K(t_i)$ in Figure 1.2) that each discrete parameter value is "correct." The residuals of the Kalman filter, based upon the "correct" model, are expected to be consistently smaller (relative to the filter's internally computed residual rms values) than the residuals of the other mismatched filters (i.e., based upon "incorrect" models) [11]. If that is true, then the MMAF algorithm appropriately weights that particular Kalman filter more heavily than the other Kalman filters. These values are used as weighting coefficients to produce a probability-weighted average of the elemental filter outputs [11]. Therefore, the state estimate ($\hat{x}_{mmaf}(t_i)$ in Figure 1.2) is actually the probabilistically weighted average of the state estimates generated by each of the K separate Kalman filters ($\hat{x}_k(t_i)$ in Figure 1.2). Testing of the three filter models was conducted for three different flight trajectories which included 2-g, 10-g, and 20-g pull-up maneuvers. Unfortunately, the residuals of the K Kalman filter did not differ from each other enough to perform the weighting function properly, and MMAF did not track well. The BM and CTR filters both performed equally well at 2-g's. The CTR filter was found to be substantially better than the BM filter for 10-g and 20-g pull-up maneuvers.

Mercier had assumed that the filter had *a priori* knowledge of the target shape and intensity profile. Singletery [42] improved the realism in the target model by developing a model in the FLIR plane which included multiple hot spots. However, he returned to the case of very benign targets. The filter did not assume *a priori* knowledge of the target size, shape, or location.

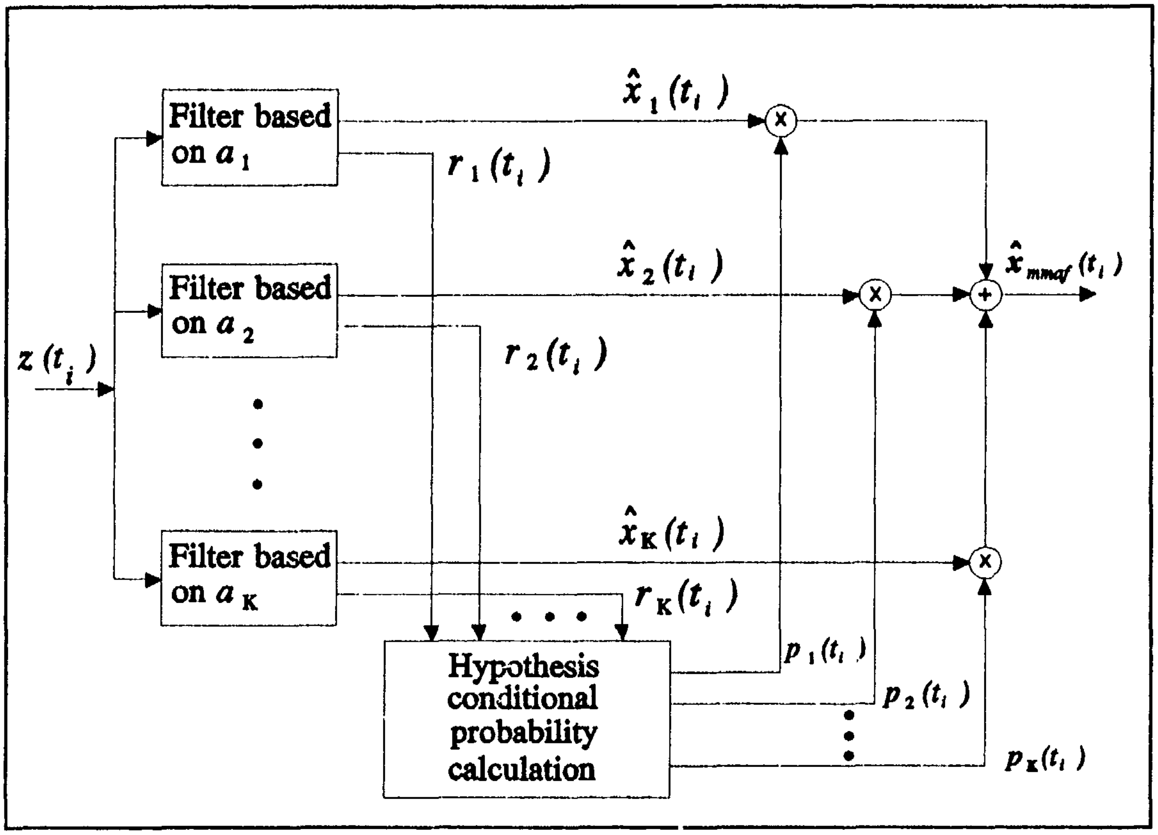


Figure 1.2 Multiple Model Adaptive Filter

A new data processing scheme (Figure 1.3) was developed which included the use of the Fast Fourier Transform (FFT) and the Inverse Fast Fourier Transform (IFFT), each of which can be produced with a lens if optical processing is used. The plan included two data paths for processing the intensity measurements $z(t_i)$. On the first path, the 8 x 8 array of intensity measurements from the FLIR are arranged into a 64-dimensional measurement vector. This measurement vector is applied to the extended Kalman filter (as in prior work). The purpose of the second path is to provide centered target shape functions to be time-averaged with previous centered shape functions in order to generate the estimated target template (***h*** in Figure 1.3) and partial derivatives of it with respect to the states (***H*** in Figure 1.3), as needed by the extended

Kalman filter. This invokes the shifting theorem of Fourier transforms. The shift theorem states that a translation of an image in the spatial domain results in a linear phase shift in the spatial frequency domain. To negate the translational effects of an uncentered target image in the spatial domain, the Fourier transform of the translated image is multiplied by the complex conjugate of the desired linear phase shift [42]. The extended Kalman filter model, in path one, which was developed by Mercier [31], was used to provide the optimal estimate of the required linear translation. The filter state estimates are used to develop the complex conjugate of the linear phase shift and provide the centered measurement functions. Before the IFFT is taken, the

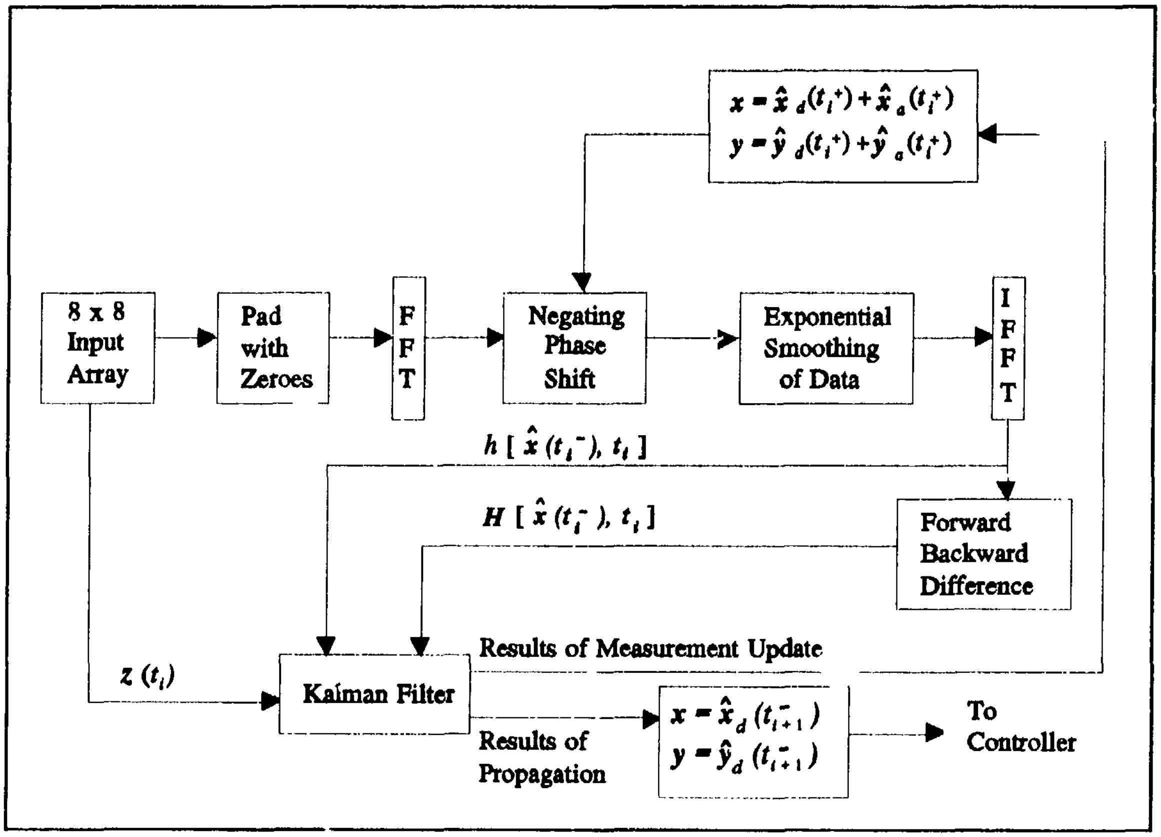

$x = \hat{x}_d(t_i^+) + \hat{x}_a(t_i^+)$
$y = \hat{y}_d(t_i^+) + \hat{y}_a(t_i^+)$
8 x 8 Input Array
Pad with Zeroes
FFT
Negating Phase Shift
Exponential Smoothing of Data
IFFT
$h[\hat{x}(t_i^-), t_i]$
$H[\hat{x}(t_i^-), t_i]$
Forward Backward Difference
Results of Measurement Update
$z(t_i)$
Kalman Filter
$x = \hat{x}_d(t_{i+1}^-)$
$y = \hat{y}_d(t_{i+1}^-)$
Results of Propagation
To Controller

Figure 1.3 Data Processing Scheme using FFT and IFFT

resulting pattern is exponentially smoothed to yield an approximation to averaging the result with previously centered frames of data, to minimize the effect of measurement noise. The result is a centered pattern with noise effects substantially reduced. Following the application of the IFFT to form the nonlinear function of intensity measurements (***h*** of Figure 1.3), the spatial derivative is used to determine the linearized function of intensity measurements (***H*** of Figure 1.3). These are both used by the Kalman filter in processing the next sampled measurement [42]. The results of this data processing scheme were inconclusive due to filter divergence problems. Despite the problems encountered with the filter, the concept was considered to have filter performance potential.

Rogers [41] continued the work of developing a Kalman filter tracker which could handle multiple-hotspots with no *a priori* information as to the size, shape, intensity, or location of the target. Moreover, he continued the application to benign target motion, as Singletary [42] had done before, in order to concentrate on the feasibility of adaptively identifying the target shape. Using digital signal processing on the FLIR data (as described above) to identify the target shape, the filter uses the information to estimate target offset from the center of the FOV, which in turn drives a controller to center the image in the FLIR plane. Algorithm improvements included replacing the Forward-Backward Difference block of Figure 1.3 with a partial differentiation operation accomplished as a simple multiplication before the IFFT block.

Rogers also considered an alternative design that used the target image ***h*** (as generated in Figure 1.3) as a template for an enhanced correlator, as shown in Figure 1.4. The position offsets produced as outputs from the correlator were then used as "pseudo-measurement" inputs to a linear Kalman filter. The improved correlation algorithm of Figure 1.4 compares the FLIR image to an estimated template instead of the previous image, as is done in the standard correlator. This tracking concept is thus a hybrid of correlation tracking and Kalman filtering [41]. Its performance was compared to the results of earlier extended Kalman filters that used the raw FLIR data as measurements [13]. Although the extended Kalman filter performed well without *a priori* knowledge of the shape and location of the intensity centroid, the improved correlator used with the linear Kalman filter outperformed the extended Kalman filter while providing reduced computational loading.

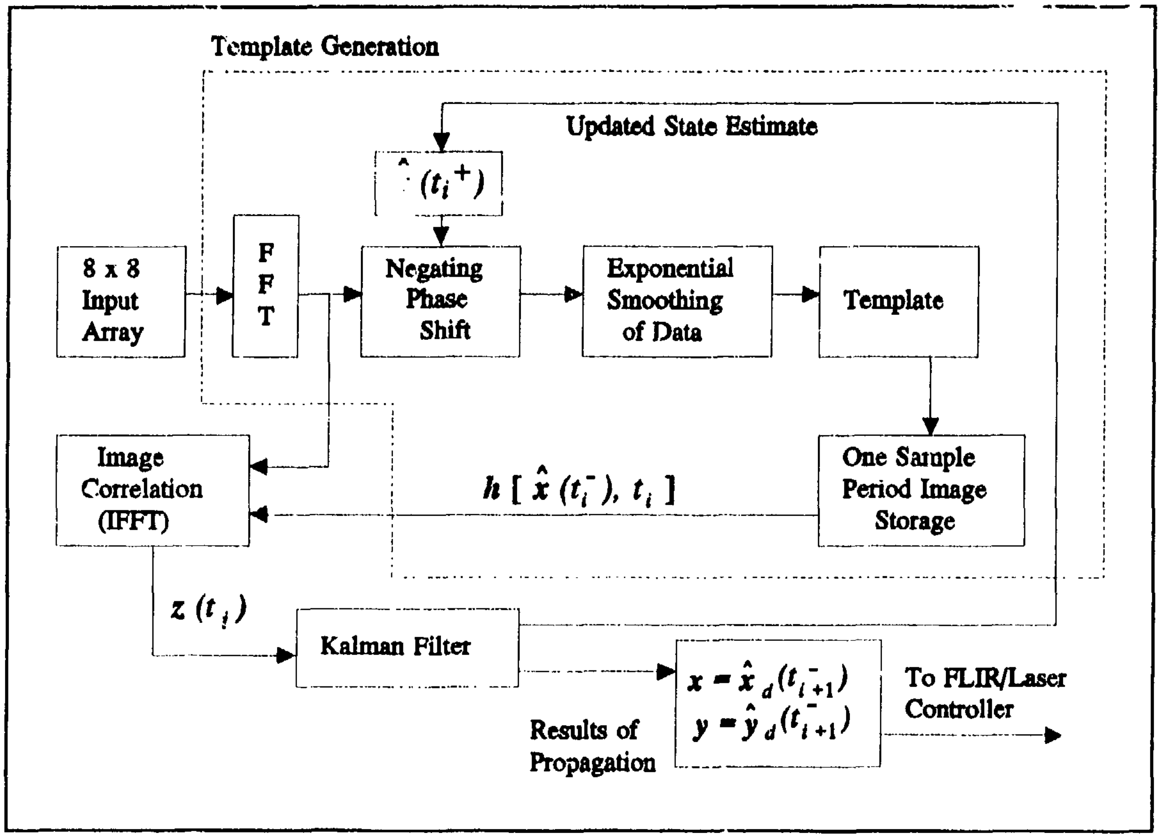


Figure 1.4 Linear Kalman Filter/Enhanced Correlator Algorithm

Millner [33] and Kozemchak [16] tested an extended Kalman filter and the linear Kalman filter/enhanced correlation algorithm against close range, highly maneuverable targets. The linear four-state filter used in the previous research was replaced by an eight-state filter consisting of position, velocity, acceleration, and atmospheric jitter states in the two coordinates of the FLIR plane (azimuth and elevation). Two target dynamics models were also developed. The target was first modeled as a first-order Gauss-Markov acceleration process, and secondly with a constant turn-rate model. Both filters performed well without *a priori* knowledge of the target size, shape, and location, using the FFT data processing method for identifying the target shape function [41,42]. However, at target maneuvers approaching 5-g's, the filter performance degraded considerably. It was noted that the tracking was substantially better when the Kalman filter dynamics model closely matched the target trajectory.

The Bayesian MMAF technique [11] was reinvestigated by Suizu [45] based on the recommendations of the previous work. The MMAF (Figure 1.2) consisted of two elemental Kalman filters. One elemental filter was tuned for benign target maneuvers and obtained sampled measurement information from an 8 x 8 pixel FOV in the FLIR plane. A second filter was tuned for dynamic maneuvers and obtained sampled measurement information from a 24 x 24 pixel FOV in the FLIR plane. The technique allowed the MMAF to maintain track on benign target trajectories up to 20-g's at a distance of 20 kilometers. The MMAF was configured for both the linear Kalman filter/enhanced correlation algorithm [41] and the extended Kalman filter. Both filtering schemes exhibited comparable rms tracking performance results, with the correlator/linear Kalman filter having smaller mean errors and larger standard deviations than the extended Kalman filter, as seen in earlier work of Rogers [41]. The state rms tracking error was on the order of 0.2 to 0.4 pixels (one pixel being equivalent to 20 μrad on a side).

The potential of the MMAF technique with the FFT processing method was continued by Loving [19]. A third filter was added to the bank of filters, tuned for intermediate target maneuvers and obtaining sampled measurement information from the 8 x 8 FOV in the FLIR plane. This MMAF showed significant performance advantages over all the previous filters. Additionally, a Maximum A Posterori (MAP) algorithm was developed and compared with the Bayesian MMAF. The MAP algorithm differs from the Bayesian MMAF of Figure 1.2 in that the MAP algorithm uses the residuals of the separate filters to select the one filter with the highest probabilistic validity, while the Bayesian MMAF uses a probability-weighted average of all filters in the bank. The Bayesian and the MAP techniques produced similar results and delivered performance that surpassed previous filters.

Netzer [37] expanded the study of the MMAF algorithm. He investigated a steady-state bias error that resulted when tracking a target exhibiting a high-g constant turn-rate maneuver. A major cause of this bias is the MMAF mistuning the *x*-direction (azimuth) while maintaining lock on the highly dynamic *y*-direction (elevation) transient for a trajectory starting horizontally and then pulling up with a high-g maneuver. This motivates the concept of individual *x*- and *y*-channel target-motion models (and tuning parameters) in the elemental filters in the MMAF, which would allow adaptive filtering for maneuvers in the *x*- and *y*-channels independently [37]. The size of the FOV was also investigated. When a target came to within five kilometers of the FLIR platform, the 8 x 8 FOV was saturated with the intensity centroid image, resulting in a loss of track. This analysis motivates a changing FOV to maintain lock for targets and also warrants the possibility of adding another Kalman filter which is tuned for extremely harsh maneuvers at close ranges. A study of the aspect ratio (AR) associated with target's intensity centroid was also accomplished to identify filter tracking characteristics for various target image functions [37].

This study used "greyscale plots" to support the analysis. A greyscale plot is a pictorial display of an image in which shading of the image is used to indicate similar parameters. In this case, the plot indicates regions of varying levels of the intensity of the filter-reconstructed target image in a 24 x 24 pixel FOV. Four different AR values of 0.2, 0.5, 5, and 10 were compared to the nominal AR of 1. The results showed that tracking was slightly impaired for images with AR as high as 5. The reduced performance is primarily along the semi-major axis of an elliptically modeled intensity centroid. Additionally, a target-decoy experiment was conducted in which a high density decoy was also located in the FOV with the target. Since the decoy was modeled with different dynamics not given to the filter, it was hoped that the filter would reject the decoy. This was not the case; the filter locked onto the hotter decoy image. This indicates that the inability of the current filter algorithm to reject this type of bright hotspot requires isolating the target image in a small FOV or some other concept to ensure tracking of the desired target.

The previous research efforts [19,37,45] used Gauss-Markov acceleration models in the development of the MMAF. Tobin [47] implemented the CTR dynamics model in another MMAF. His results showed that the Gauss-Markov MMAF exhibited smaller bias errors while the CTR MMAF gave smaller steady state standard deviation errors; both filters had comparable rms errors. Motivated by earlier research [37], he also developed an 8 x 24 pixel FOV for both the x- and y-directions of the FLIR image plane to be used with filters designed to anticipate harsh target accelerations in a specific direction (along which the longer side of the FOV would be oriented). The results showed that the filter maintained lock on a target during a highly dynamic maneuver in the y-direction while maintaining substantially better steady state bias performance in the benign x-direction.

Leeney [17] expanded the previously used Gauss-Markov truth model by incorporating bending vibrational states. The elemental filters in the MMAF were not modeled with this information through explicit state variables, but performed well up to a 10-g maneuver. A performance investigation was also conducted as to the effects of increasing the measurement update rate from the previously used 30 Hz to 50 Hz. The sampling rate of 50 Hz showed a minor performance improvement, but also increased the computational loading because of the higher rate. A preliminary study was also done on replacing the 8 x 24 pixel FOV in the x- and y-directions [47] on the FLIR plane with a single 8 x 24 pixel FOV, which is also known as the rotating rectangular-field-of-view (RRFOV). The idea was to align the long side of the rectangular FOV with an estimate of the acceleration vector. The higher precision velocity estimate was actually used instead of the noisier acceleration estimate, and it was assumed that the acceleration direction would be essentially orthogonal to the current velocity vector direction. Additionally, the five elemental Kalman filters in the MMAF bank would be reduced to four by using this FOV rotation scheme. The results were not conclusive, but the insight provided motivation to continue the study.

The RRFOV research was continued by Norton [38]. He discovered that the appropriate choice of the filter dynamics driving noise strength Q dictated the filter's response to a high-g jinking maneuver, and that the size of the FOV could be reduced to an 8 x 8 pixel rotating FOV, also known as the rotating square field of view (RSFOV). His investigation showed that a non-rotating square FOV could provide good performance, but that the dynamics noise strength Q matrix value must be large in the elements corresponding to the direction of the acceleration vector. A mathematical matrix transformation was developed which "rotated" the Q matrix to keep the larger values aligned with the acceleration vector. A study of both the rotating FOV and rotating the Q matrix provided advantages and disadvantages for each method. Both methods are affected by the tuning parameters used to represent the rms level of acceleration of the target, which also contributes to error biases. The rotating FOV improves the x-direction (azimuth) estimation for dominant y-direction (elevation) dynamics from previous MMAF algorithms (on pull-up maneuvers), but does not improve y-direction estimation for dominant y-direction dynamics. Rotating the Q matrix adaptively improves estimation of both x- and y-directions and improves the jink maneuver error transients, but is dependent on the orthogonality of the velocity and acceleration vectors and proper initial tuning parameters. The conclusion was that both methods employed together provide the ability to adjust filter characteristics to differentiate between harsh and benign dynamics in any orientation of target acceleration (rotating Q) while at the same time maintaining appropriate view resolution in the directions of both benign and harsh dynamics (rotating FOV). Therefore, the combination allows for tracking highly maneuvering targets without sacrificing the resolution provided by the smaller RSFOV [38].

The research up to this point was primarily directed towards tracking aircraft and missiles from a ground-based FLIR plane. Rizzo [40] initiated research on a space-based platform which could track targets using the same filtering techniques. Since the linear Kalman filter/enhanced correlator algorithm had proven to be computationally more efficient than the extended Kalman filter, it was chosen as the system filter. The plume "pogo" (oscillation) phenomenon of a missile in the boost phase of flight was modeled in the truth model and in one of two filters used for the analysis. The pogo was modeled as a second-order Gauss-Markov process, and applied in the direction of the missile velocity vector. The plan was to go adaptive on the pogo states using the MMAF algorithm, treating the pogo amplitude and oscillation frequency as uncertain parameters.

Although the elemental filters were developed, no MMAF performance was accomplished, due to elemental filter performance difficulties.

Three rotation schemes were also developed and tested. The first scheme, referred to as the rotating field-of-view (RFOV), involved using the 8 x 8 FOV filter and aligning a single axis of the FLIR plane with the estimated velocity vector of the target; therefore one of the coordinate axes of the FOV would stay aligned with the oscillation of the plume. The second scheme, referred to as the diagonal rotating field-of-view (DRFOV), used the 8 x 8 FOV with the diagonal aligned with the oscillation of the plume. The motivation behind this scheme is that the 8 x 8 FOV is oriented in such a fashion will be able to "see" more of the target's intensity image, thus enabling the sensor to obtain more measurement information [40]. The third tracking scheme was the rotating rectangular field-of-view (RRFOV) algorithm developed from previous research [17,47]. The RFOV, DRFOV, and the RRFOV algorithms [37] were tested along with the non-rotating field-of-view (NRFOV) filter. The NRFOV is the standard tracker used in previous studies [17,37,47]. The DRFOV scheme was shown to be superior to the other three tested for providing enhanced tracking of a missile hardbody whose plume is undergoing a pogo phenomenon.

The eight-state filter (without pogo states; two target position states, two target velocity states, two target acceleration states, and two atmospheric jitter states) and the ten-state filter (with pogo states) surfaced a problem that may have gone unnoticed in previous work. Following tuning of the filters with the twelve-state truth model, it was discovered that the eight-state filter outperformed the ten-state filter. An investigation into the cause of the irregularity revealed that there was a serious observability problem in the both filters. The affected states were velocity and acceleration. A recommendation was made to remove the acceleration states in the ten-state filter, and to model the velocity states in this new eight-state filter as a first-order Gauss-Markov process.

Eden [8] resumed the research of the space-based FLIR platform. The scope of the tracking problem was expanded by requiring the filter to track the hardbody of the missile rather than just the intensity centroid of the FLIR. Since the FLIR could not supply the needed information about the hardbody location relative to the image center-of-intensity to the Kalman filter, another measurement source was developed. Under the advisement of the Phillips Laboratory, the new measurement source was identified as a low-energy laser. The laser actively acquires measurement data while the FLIR obtains its measurement information passively. This scheme calls for a six-state Kalman filter (consisting of two position states, two velocity states, and two atmospheric jitter states) to provide both a position vector and a velocity vector estimate for the target plume. The low-energy laser is scanned along this estimated velocity vector from the target plume image intensity center to intercept the hardbody. The hardbody is modeled as a rectangle with binary reflectivity. When the low-energy laser (modeled with a beam width of 2.75 meters at the target) illuminates the hardbody, the reflection is received by a low-energy laser sensor on the platform. This speckle information is provided to a single-state Kalman filter which estimates the distance between the center-of-mass and the center-of-intensity along the velocity vector direction. The center-of-mass is defined as the midpoint of the scan across the hardbody if the centerline of the laser beam crosses the aft end of the missile and the top (nose) of the hardbody, or if the laser beam crosses the aft end and one of the sides of the hardbody. The results of the laser scan show that the interception of the laser with the hardbody occurred only 10-20% of the time. This low ratio of hitting the target was attributed to the six-state filter being tuned for estimating only the intensity centroid location on the FLIR plane and not for precise velocity estimation. Since the velocity vector must be accurately estimated for active illumination of the target to be a viable concept, it was recommended that the filter also be tuned for accurate velocity estimates.

Tracking the center-of-mass of a missile hardbody using FLIR measurements and low-energy laser illumination was further investigated by Evans [9]. He surmised that the tracking error, represented by a straight line between the estimated target center-of-mass and the true center-of-mass [8], could provide more insight if it were separated into the *x*- and *y*- (azimuth and elevation) components, or into along-track and across-track (2-d perpendicular axes of the hardbody) components. Evans proposed the latter method would provide better information relative to the principle axes directions of the error phenomenon. An eight-state filter was developed by augmenting Eden's six-state filter [8] with two additional bias states used to estimate the hardbody center-of-mass [9]. A comparison between the eight-state filter and Eden's one-state filter used in conjunction with the six-state FLIR filter, resulted in negligible difference in performance. Evans' analysis of the eight-state filter's error statistics showed that the tracking error is much greater in the along-track direction than in the across-track direction, and thus the separate one-state filter and six-state FLIR filter performed as well as the all inclusive eight-state filter.

Aside from investigating the tracking error statistics, Evans enhanced Eden's 2-d hardbody model (which treated reflectivity as a binary on/off function) with a 3-d hardbody reflectivity model to provide increased realism in the simulation. Two reflectivity functions, cross-sectional and longitudinal, were defined based upon empirical data obtained from a radar return off a 20 x 249 inch cylinder with hemispherical endcaps, rotated longitudinally in the plane of the radar source [10]. As shown in Figure 1.5, the cross-sectional and longitudinal reflectivity functions were incorporated into Eden's rectangular hardbody model as 29 discrete weighted line segments along the longitudinal axis of the hardbody.

Evans also found that the sensitivity level of the low-energy sensor is a factor in determining the reflectivity received at the sensor [9]. The sensitivity level represents a threshold below which the reflected return is indistinguishable from sensor noise. A sensitivity factor, μ, is incorporated in the simulation to define the appropriate sensitivity level required to detect a hardbody's return as well as represent the physical limitations of the sensor.

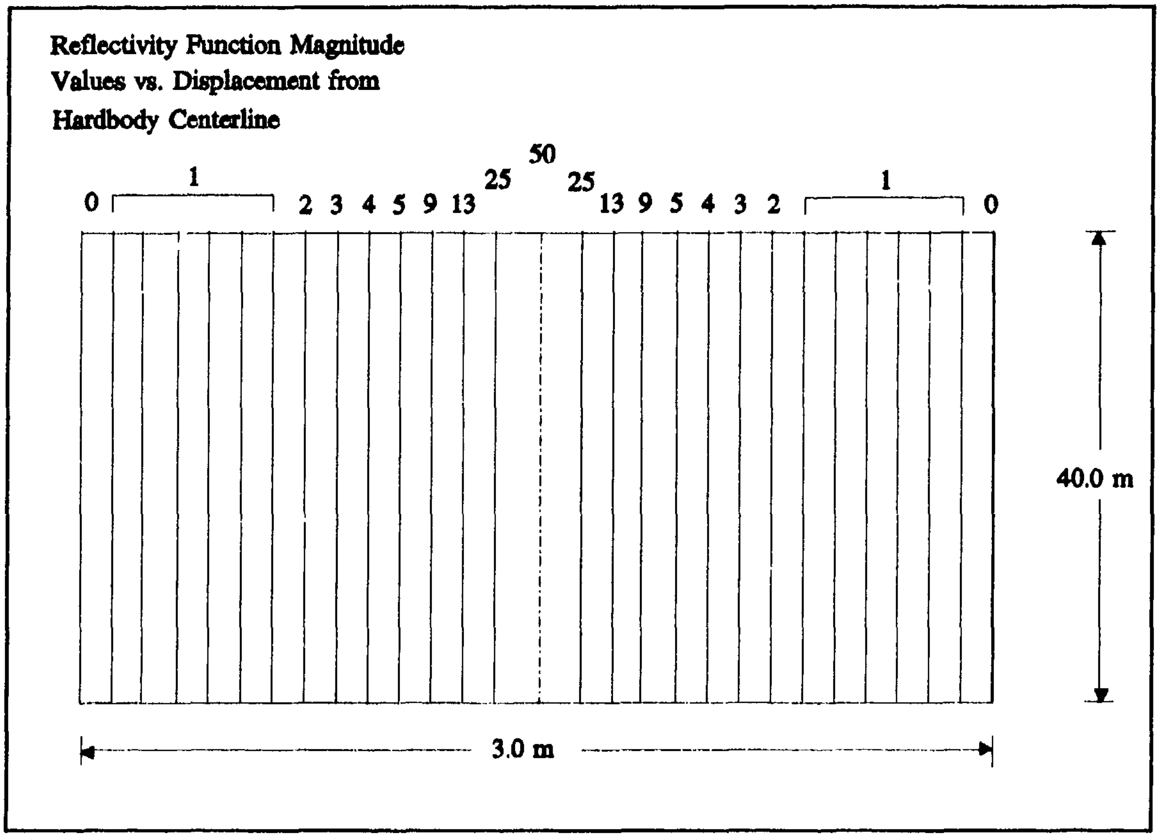


Figure 1.5 Discrete Implementation of Cross-Sectional Reflectivity Function

Performance data collection from the eight-state filter and one-state/six-state filter combination hinged upon the successful illumination of the hardbody by the low-energy laser. Evans was faced with a low target intercept rate (10% - 20%), which inhibited any useful error analysis of the center-of-mass filters. Realizing this, Evans generated an ad hoc technique of offsetting the low-energy laser scan relative to the FLIR estimated velocity vector and "sweeping" the scan across the hardbody, thus providing constant hardbody illumination information. However, the "sweep" is not an optimal tool and should only be used to test the center-of-mass filters in the simulation [9]. Both the 3-d reflectivity hardbody model and laser sweep were employed to evaluate the performance of the eight-state filter and one-state/six-state filter combination center-of-mass estimators.

Herrera continued to investigate the use of laser returns to determine the offset between the hardbody and the plume intensity centroid, however, he used the information contained in the Doppler spectra of the returns as opposed to the speckle reflectance magnitude [14]. Experiments had shown that the laser speckle return of a solid-propellant rocket motor is of the same magnitude as that of the hardbody as a result of the metallic particles present in the propellant [2]. The returns from the plume can cause a non-negligible bias in the intensity centroid to center-of-mass offset estimate of 25 to 30 meters, up to 90% of the times a laser scan is successful. This tendency was not incorporated into the simulations completed by Eden and Evans [8,9]. Herrera first showed that, as suspected, a bias, dependant on plume speckle return, existed in the offset estimates using the one-state/six-state filter combination utilized by Eden and Evans.

Herrera proposed that the two types of information in the Doppler frequency spectra, magnitude of frequency shift and spread of the return spectrum, could be used to obtain a finer discernment of the plume/hardbody interface. This proposal was based on the fact that the spectral content of the hardbody and plume returns exhibit very different Doppler characteristics that should be readily distinguishable. His approach to using this information was not to simulate the Doppler phenomenon itself, but to simulate the quality of the returns provided as measurements to a single-state linear Kalman filter that estimates the offset. Herrera simulated the quality of the low energy laser return as a function of laser wavelength and signal-to-noise ratio, and simulated a specified probability of no Doppler information at a given sample time due to either the plume and hardbody spectra being indistinguishable or the low-power laser beam missing the target body [14].

To prove the utility of using Doppler spectra, Herrera developed a one-state Doppler filter to replace the one-state offset estimator used by Eden and Evans [8,9]. He maintained the same independent filter structure as used before, which utilized a six-state filter (four target dynamics and two atmospheric jitter states) in conjunction with the offset filter. He also developed a two-state Modified MAP MMAF that incorporated both the speckle return and the Doppler return measurements. Both configurations successfully showed that the Doppler return information allowed more accurate determination of the hardbody/plume interface. The one-state filter based only on Doppler measurement data delivered unbiased estimates of the offset, and in the case of the two-state adaptive filter based on both Doppler and speckle information, use of Doppler spectra permitted accurate calculation of the bias in the offset measurement from the speckle return [14].

1.3 Thesis Objectives

Although the past three theses have concluded that LEL measurements can be successfully used in the tracker, they have been based on a simplified target dynamics model without acceleration or pogo states in the simulation of the real world environment. It is very difficult to estimate the position of the hardbody itself using FLIR measurements alone because the FLIR only detects IR radiation from the plume. Two major problems must be addressed while designing an accurate tracking filter. The first is determining the observability problems in the filter, and the second is finding the location of the hardbody itself.

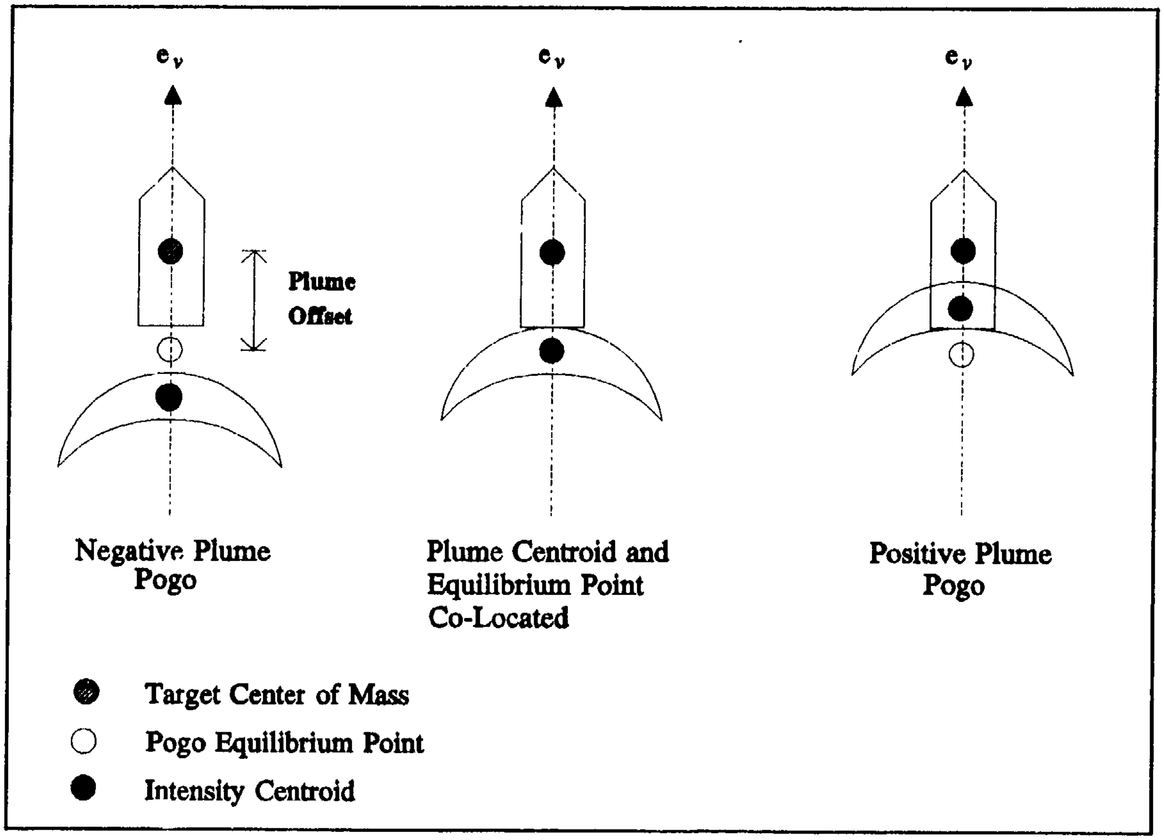


Figure 1.6 Pogo Phenomenon

Apparent motion of the plume centroid in the FLIR plane (as modeled in the filter) can be caused by three major effects: motion of the target, the pogo effect, and jitter in the measurements caused by the atmosphere. The current filter design does not have the capability to observe the differences between the three effects fully . The next phase of research will finish identifying the specific observability problems discovered by Rizzo [40], and particularly will determine if there are observability problems associated with jitter and/or pogo. Next, a series of studies must be completed to incorporate the pogo effect into the model. The pogo effect, depicted in Figure 1.6, is the phenomenon in which the centroid of the plume oscillates along the target velocity vector about an offset distance from the hardbody center-of-mass.

Once the observability problems are identified, design of a complete filter can be pursued. A complete filter model will include all target dynamics states, the measurements from the FLIR, and the measurements from the LEL. The measurements from the LEL will provide information on the missile hardbody so that the filter may estimate hardbody location. Note that, due to the observability difficulties identified by Rizzo [40], the last three theses have not included pogo in either the truth or filter models. Therefore, the robustness of filters including LEL measurements in the in the presence of the pogo effect has not been validated. In addition, the structure of the filters including LEL measurements was such that plume intensity centroid location and hardbody offset were determined independently and the center-of-mass was calculated outside the filter based on the estimates. This research will attempt to estimate the hardbody center-of-mass directly using a single filter.

1.3.1 Observability Gramian Analysis. The first main objective in the this thesis will resume the tests to determine what is causing the observability problems in the filter.

Observability gramians and eigenvalues for six different filter models will be investigated. It is suspected that the observability gramian matrix eigenvalues associated with either the pogo states, the jitter states, or both, are small relative to the other eigenvalues. If one set of eigenvalues is small in comparison to the other, then oscillation of the plume intensity centroid relative to the hardbody along the velocity vector would be dominated by a single effect and the filter would not be able to distinguish between the jitter and pogo effects by using only measurements on the rocket plume (in other words--only FLIR measurements).

The first stochastic observability tests will be performed on two linear Kalman filters without pogo states. One filter will have 12 states including a full atmospheric jitter model of 6 states, 6 target dynamics states (position, velocity, and acceleration in the X and Y directions). The other filter will be an eight-state filter that has an approximated atmospheric jitter model (only two jitter states), and six target dynamics states.

Next, the same models but without the acceleration states will be studied. Comparison of the observability tests on these 10-state and 6-state filters with the previous 12 and 8-state filters respectively, should provide insight into whether there is a fundamental problem with how the full atmospheric jitter model and acceleration states are affecting each other. These first two sets of tests should also indicate whether there is some interaction between the acceleration states and jitter states that may be causing degradation of filter performance.

Observability will also be investigated for an alternate eight-state filter model that excludes the acceleration states. This model will include four target dynamic states, the two pogo states and the two-state jitter model. A velocity vector at a constant angle of 45 degrees will be simulated to implement the pogo states for these studies. These tests can be compared to results from the same model with the pogo states removed to find any difficulties stemming from pogo/jitter blending. The study will be repeated for the same filter models outlined above but with a general angle.

The outcomes of the stochastic observability tests will also be analyzed for observability difficulties of jitter itself because previous research shows $\hat{x}(t_{i+1}^-)$ errors are so much worse than $\hat{x}(t_i^+)$ errors [8,9,14,40]. The differences in error performance (varying from a factor of two to over an order of magnitude) seem to be due to problems in estimating jitter. This research will attempt to determine whether there is a fundamental problem with the jitter model itself that causes such poor estimation of atmospheric disturbances over the filter propagation cycle.

If the studies above confirm the suspicion that pogo and jitter states are difficult to distinguish, then one may conclude that measurements from the FLIR of the plume intensity centroid alone cannot provide the desired tracking performance of the missile hardbody due to limited estimation accuracy. Inclusion of additional measurements, such as laser Doppler and/or speckle, should increase filter performance for a number of reasons. Laser measurements provide information on the missile hardbody itself as opposed to just the missile plume. Furthermore, laser measurements should enhance observability of the pogo and jitter states. By "knowing" where the hardbody is in relation to the plume, the filter will be able to distinguish the fundamental difference between jitter and the pogo phenomenon. Atmospheric jitter will be manifested as oscillations of both the hardbody *and* plume in certain frequency ranges. Pogo should be differentiable as a *relative* oscillation between the hardbody and plume with a different characteristic power spectral density (PSD). A more thorough descriptions of the truth and filter models are contained in Chapters III and IV.

1.3.2 Sensitivity Analysis of Single Filters. In order to verify the results of the observability studies completed in the previous section, a series of 16 Monte Carlo performance analyses will be performed using a software package called Multimode Simulation for Optimal Filter Evaluation (MSOFE) [6]. Table 1.2 shows the combinations of truth versus filter models

Table 1.2 Monte Carlo Analysis Combinations

	TRUTH MODEL					FILTER MODEL			
SIM	N	DYN	POGO	ACCEL	JITTER	JITTER	DYN	POGO	N
1	4	4					4	2	6
2	6	4			2		4		4
3	6	4			2		4	2	6
4	6	4			2	2	4		6
5	6	4			2	2	4	2	8
6	6	4	2				4	2	6
7	8	4	2		2		4		4
8	8	4	2		2		4	2	6
9	8	4	2		2	2	4		6
10	8	4	2		2	2	4	2	8
11	8	4	2	2			4		4
12	8	4	2	2			4	2	6
13	10	4	2	2	2		4		4
14	10	4	2	2	2		4	2	6
15	10	4	2	2	2	2	4		6
16	10	4	2	2	2	2	4	2	8

to be investigated and the states each model contains. A subset of the filter models studied in the previous section will be compared, via 10-run Monte Carlo analyses, against various truth models that incorporate different combinations of states. The number of analyses has been reduced to include only the combinations that should best provide insight into how the observability difficulties are affecting filter performance. Monte Carlo analysis should indicate how estimatable the individual states are for each combination of filter and truth model.

The first five runs shown in Table 1.2 investigate performance of filter models with combinations of the approximated atmospheric jitter model and pogo states in comparison to a truth model with no pogo phenomenon modeled. The second five runs analyze the same filter models against a system model that includes the pogo states. These first two sets of Monte Carlo analyses should help to identify the relative effects of the jitter and pogo states in the filter model when presented with different real-world simulations. It is predicted the effects due to jitter are dominant in the filter model and addition of the pogo states may actually degrade filter performance due to potential observability difficulties. It is further suspected that filter performance against a system without pogo modeled will be better than performance against a system that incorporates the pogo phenomenology.

The last six sets of Monte Carlo analyses utilize two different truth models with target dynamic acceleration states included. One truth model (10-state) incorporates jitter states and the other (8-state) does not. The performance of four different filter models, with and without jitter and/or pogo states, are analyzed against the two truth models. The last sets of Monte Carlo runs will provide insight into what combinations of filter states provide the best performance when the target acceleration is present. Since target acceleration is extremely difficult to estimate in the long range ballistic missile problem, it is suspected that performance of all filters will be degraded by adding the target acceleration states to the truth model. In addition, it is doubtful that including the acceleration states in the *filter* model will alleviate the problem. In fact, performance may be degraded by integrating the poorly estimated states.

An additional benefit of using the MSOFE package for the aforementioned Monte Carlo analyses, is that it may (or may not) provide independent verification of results obtained from the AFIT software. Since the simulations in MSOFE will use continuous-time dynamics models and will be tested with independent software code, any major differences in results may shed light on the particular truth model simulation in the existing AFIT software.

1.3.3 Incorporation of Laser Measurements into Elemental Filters. Based on insight from previous research, it is expected that a truth model that incorporates the six-state jitter model, the pogo phenomenon and four target dynamics states will be used for this research. It is desired that the filter including the pogo states that performed best in the previous analysis be used to develop the optimal filter that directly estimates hardbody center-of-mass location.

The results from the previous step will be combined with the last three years' research to design and evaluate a filter that incorporates the low energy laser measurements with no intentional decoupling of the offset state from the other states. This research will incorporate the Doppler measurement model developed by Herrera [14]. The reason this model was chosen is that it delivers offset measurements that are not corrupted by any bias due to plume speckle reflectance. The Doppler measurement model is also a more realistic simulation of the real world since it incorporates the 3-dimensional reflectivity model developed by Evans [9]. To establish the validity of the new filter, the simulations will be chosen such that the highest quality of LEL measurements are produced. Therefore, initial tests of the filter will be under the most benign conditions (i.e. low probability of miss, high signal-to-noise ratio, and short LEL wavelength as described by Herrera).

The key difference of the new elemental filter will be the inclusion of the pogo dynamics model in the design, under the assumption that its undamped natural frequency and expected rms amplitude are known. Another key feature of the elemental filter will be the revision of the filter states such that the target dynamics states describe the hardbody center-of-mass, as opposed to the plume intensity centroid. Performance of the filter will be analyzed via Monte Carlo simulations in the AFIT software. The intent of this objective is to build a single elemental filter which can eventually be incorporated into a MMAF structure, allowing that adaptive structure to identify the actual undamped natural frequency and rms amplitude of the real-world pogo phenomenon.

1.3.4 Multiple Model Adaptive Filter Design and Analysis. After a single, optimal filter structure that utilizes all both the FLIR and LEL sensor measurements is established, the preliminary work for a MMAF can begin. The MMAF is an algorithm that incorporates several filters running simultaneously. The algorithm monitors the errors from each filter and selects the solutions from the filters with the lowest errors at any instant in time. A MMAF allows the tracker to use a filter that is tuned for the exact conditions in existence, as opposed to a general filter tuned for a wide variety of conditions. A block diagram of the structure of a MMAF is shown in Figure 1.2. In Figure 1.2, $\hat{x}_K$ is the filter estimate, r_K is the error, or residual, and p_K denotes the probability weighting factor that determines which individual filter solutions are used at the particular instant in time.

The first step in the design of the MMAF is to identify the elemental filters that will be used in the filter bank. The parameter values, a, that the filter is based on will be varied and the filter tuned for maximum performance based on those different assumed values. Since the design of the elemental filter described in the previous section was dependent on prior knowledge of the pogo conditions, frequency and amplitude of the pogo oscillation will be the parameters chosen for variation in the MMAF structure. The tuning process will result in a series of filters with identical structure, but each tuned for optimal performance under different target conditions (specifically, pogo conditions). A baseline performance for each of the above filters must be established through Monte Carlo analysis before moving to the next step. This research actually accomplishes the design of one such elemental filter, as the essential first step to the design and analysis of such an MMAF.

Once the individual filters have been selected, they may be incorporated into the adaptive algorithm structure. Then, the entire MMAF could eventually be evaluated via Monte Carlo sensitivity analysis. This analysis could be compared to the baseline performances previously established with the elemental filters to determine adequacy of the adaptive parameter estimation. The observability tests described in the first objective, combined with the tuning accomplished for the elemental filters, may provide the insight to develop alternate adaptive algorithms. It is expected that the performance of the MMAF or MMAFs will eventually be better than any filters previously designed.

1.4 Thesis Overview

This chapter has described the past AFIT research in the use of Kalman filters to track airborne targets. It has also introduced the specific problem research that this thesis will be directed toward solving, and the approach and objectives have been set in the process. Chapter II contains a basic discussion of linear Kalman filter theory and the multiple model adaptive algorithm. It also contains a description of the coordinate frames used in the thesis and how they relate to the simulated tracking scenario. The truth models used in the thesis are presented in Chapter III, as well as specifics on how the various FLIR and LEL measurements are modeled. Chapter IV contains a development of the filter models. Chapter V describes the procedures and results obtained from the preliminary observability studies, including both the observability gramian tests and the MSOFE Monte Carlo simulations. Chapter VI will portray results from the simulations completed with the AFIT software. Finally, Chapter VII lays out the conclusions and recommendations for future research efforts.

II. Filtering Techniques and Reference Coordinate Systems

2.1 Introduction

This chapter describes the technical filtering theory upon which the thesis is based. A brief review of the linear Kalman filter (LKF) is given. Since the new filter designs incorporating the plume pogo states and the low energy laser measurements require a nonlinear measurement update, the extended Kalman filter equations for a nonlinear update are also reviewed. This chapter is meant only as a review and the reader is referred to the source literature for a more thorough development of the topics [21]. The elemental filters are based on the theory presented in the Kalman filter (KF) section, but the structure of the Multiple Model Adaptive Filter (MMAF) is also discussed briefly in this theory chapter. Although this thesis did not develop an MMAF algorithm, the original intent was eventually to incorporate the elemental filter that was developed into a MMAF, so the theory is presented here for completeness

In addition, the author has included a section that explains the reference coordinate systems utilized throughout the thesis. The reference system used for the real world three dimensional simulation space is presented first, followed by all the other reference systems and appropriate coordinate transformations necessary to accommodate the appropriate sensors and corresponding filter algorithms.

None of the theory contained in this chapter has changed and all of it has been developed in previous theses. Consequently, the author has taken the liberty of borrowing heavily from the excellent presentations made by his predecessors [8,9,14,40].

2.2 Kalman Filter Theory

2.2.1 Introduction. A Kalman filter (KF) is "an optimal recursive data processing algorithm"[21]. The KF uses all available information to estimate the current values of variables of interest (the state vector) of a system. The KF arrives at its estimate through *a priori* knowledge of the system and measurement device dynamics; statistics of noise corruption, uncertainties, and measurement errors; system noises and uncertainties; and initial conditions. The estimate provided by the KF is *the* optimal estimate if the system can be modeled as linear and driven by white Gaussian noise. The optimal estimate will be better than any estimate available from any single measuring device. The important feature about the KF is that it not only directly models the variables of interest, but also models errors so that it provides estimates of how accurate it "thinks" the state estimate is (through the filter covariance). The following is a simplified version of the development presented by Maybeck [21]. For a rigorous development of the Kalman Filter theory and MMAF algorithm, the reader is referred to Maybeck's *Stochastic Models, Estimation, and Control, Vol. 1* and *Vol. 2*. The sections describing the Kalman filter were reproduced from Evans and Herrera [9,14] and the MMAF section is repeated in whole from Herrera [14].

The *a priori* statistics of the mean and covariance provided to the filter as initial state conditions are defined by:

$$E\{\mathbf{x}(t_o)\} = \hat{\mathbf{x}}_o \tag{2-1}$$

$$E\{[\mathbf{x}(t_o) - \hat{\mathbf{x}}_o][\mathbf{x}(t_o) - \hat{\mathbf{x}}_o]^T\} = \mathbf{P}_o \tag{2-2}$$

where the notation (^) indicates an estimated value, and $E\{\ \}$ is the expectation, or ensemble average, of the possible outcomes. The Kalman filter receives measurements at a prescribed sample rate and propagates the state conditioned upon the measurement time history $Z(t_i)$, given as:

$$Z(t_i) = \begin{bmatrix} z(t_1) \\ \cdot \\ \cdot \\ \cdot \\ z(t_i) \end{bmatrix} \tag{2-3}$$

where $z(t_j)$ is the measurement data available at sample time t_j. Then the conditional mean and covariance of the state variables are given by:

$$\hat{x}(t_i^+) = E\{x(t_i) \mid Z(t_i) = Z_i\} \tag{2-4}$$

$$P(t_i^+) = E\{[x(t_i) - \hat{x}(t_i^+)][x(t_i) - \hat{x}(t_i^+)]^T \mid Z(t_i) = Z_i\} \tag{2-5}$$

where Z_i is a specific realization (observed set of values) of the measurement history $Z(t_i)$.

2.2.2 Linear Kalman Filter. The simplest form of Kalman filter, the linear Kalman filter, is developed in this section. In order to develop the linear KF, a mathematical model of the system dynamics must be established and measurements must be available. A system is generally modeled with a set of linear state differential equations of the form:

$$\dot{x}(t) = F(t)x(t) + B(t)u(t) + G(t)w(t) \tag{2-6}$$

where

$F(t)$ = homogeneous state dynamics matrix
$x(t)$ = vector of states of interest
$B(t)$ = control input matrix
$u(t)$ = deterministic control input vector
$G(t)$ = driving noise input matrix
$w(t)$ = white Gaussian driving noise vector

The mean of the white Gaussian driving noise vector is:

$$E\{w(t)\} = 0 \tag{2-7}$$

and the noise strength is $Q(t)$:

$$E\{w(t)w(t+\tau)^T\} = Q(t)\delta(\tau) \tag{2-8}$$

The equivalent discrete-time system model of Equation (2-6) is needed to implement the algorithm on a digital computer. The general form of the discrete-time state space form (denoted by the *d* subscript) of that model is given by:

$$x(t_{i+1}) = \Phi(t_{i+1},t_i)x(t_i) + B_d(t_i)u(t_i) + w_d(t_i) \tag{2-9}$$

where

$\Phi(t_{i+1},t_i)$ = the n x n system state transition matrix that satisfies the differential equation and initial condition:

$$\frac{d[\Phi(t,t_i)]}{dt} = F(t)\Phi(t,t_i) \tag{2-10}$$

$$\Phi(t_i,t_i) = I \tag{2-11}$$

and where

$x(t_i)$ = discrete-time vector of states of interest

$B_d(t_i)$ = discrete-time control input matrix

$u(t_i)$ = discrete-time deterministic control input vector

$w_d(t_i)$ = discrete-time independent, white Gaussian noise process with mean and covariance statistics defined as:

$$E\{w_d(t_i)\} = 0 \tag{2-12}$$

$$E\{w_d(t_i)w_d^T(t_j)\} = \begin{Bmatrix} Q_d(t_i) & t_i = t_j \\ 0 & t_i \neq t_j \end{Bmatrix} \tag{2-13}$$

with

$$\mathbf{Q}_d(t_i) = \int_{t_i}^{t_{i+1}} \mathbf{\Phi}(t_{i+1},\tau)\mathbf{G}(\tau)\mathbf{Q}(\tau)\mathbf{G}^T(\tau)\mathbf{\Phi}^T(t_{i+1},\tau)\,d\tau \tag{2-14}$$

The Kalman filter incorporates measurement information from external measuring devices to improve its estimate of a desired state. The discrete-time (sampled data), linear measurement model is of the form:

$$\mathbf{z}(t_i) = \mathbf{H}(t_i)\mathbf{x}(t_i) + \mathbf{v}(t_i) \tag{2-15}$$

where

$\mathbf{z}(t_i)$ = m-dimensional measurement vector at sample time t_i
$\mathbf{H}(t_i)$ = state observation matrix
$\mathbf{x}(t_i)$ = vector of states of interest
$\mathbf{v}(t_i)$ = white Gaussian measurement noise

The discrete white Gaussian measurement noise $\mathbf{v}$ is independent of both $\mathbf{x}(t_o)$ and $\mathbf{w}$ for all time, and has a mean and covariance, $\mathbf{R}$, given by:

$$E\{\mathbf{v}(t_i)\} = \mathbf{0} \tag{2-16}$$

$$E\{\mathbf{v}(t_i)\mathbf{v}^T(t_j)\} = \begin{Bmatrix} \mathbf{R}(t_i) & t_i = t_j \\ \mathbf{0} & t_i \neq t_j \end{Bmatrix} \tag{2-17}$$

The Kalman filter propagates the state conditional mean and its covariance from the instant in time immediately following the most recent measurement update, t_i^+, to the instant in time immediately preceding the next measurement update, t_{i+1}^-, by numerical integration of the following equations:

$$\dot{\hat{\mathbf{x}}}(t/t_i) = \mathbf{F}(t)\hat{\mathbf{x}}(t/t_i) \tag{2-18}$$

$$\dot{P}(t/t_i) = F(t)P(t/t_i) + P(t/t_i)F^T(t) + G(t)Q(t)G^T(t) \tag{2-19}$$

where the notation $\hat{x}$ (t/t_i) denotes optimal estimates of x at time t, conditioned on measurements through time t_i, and with initial conditions:

$$\hat{x}(t_i/t_i) = \hat{x}(t_i^+) \tag{2-20}$$

$$P(t_i/t_i) = P(t_i^+) \tag{2-21}$$

where $\hat{x}$ (t_i^+) and $P(t_i^+)$ are the results of the previous measurement update cycle. At time t_o, $\hat{x}_o$ and P_o from Equations (2-1) and (2-2) are used to initialize the first propagation.

That update cycle, when a measurement becomes available at time t_i, is based on the following update equations:

$$K(t_i) = P(t_i^-)H^T(t_i)[H(t_i)P(t_i^-)H^T(t_i) + R(t_i)]^{-1} \tag{2-22}$$

$$\hat{x}(t_i^+) = \hat{x}(t_i^-) + K(t_i)[z(t_i) - H(t_i)\hat{x}(t_i^-)] \tag{2-23}$$

$$P(t_i^+) = P(t_i^-) - K(t_i)H(t_i)P(t_i^-) \tag{2-24}$$

where $K(t_i)$ is the time-varying Kalman filter gain matrix that assigns "weights" to the new information (consisting of the difference between the actual measurement and the filter's prediction of the measurement, $H(t_i)\hat{x}(t_i^-)$, as seen in Equation (2-23)) based on known measurement noise statistics and filter-computed covariances.

In some instances (as with the case when pogo is included in the elemental filter), the discrete-time measurement update is a known nonlinear function of the state vector. In such cases, the following nonlinear extended Kalman filter update model is used in place of Equation (2-15). The measurements are modeled as:

$$z(t_i) = \boldsymbol{h}[\boldsymbol{x}(t_i), t_i] + \boldsymbol{v}(t_i) \tag{2-25}$$

where $\boldsymbol{h}[\cdot,\cdot]$ is a known vector of functions of state and time, and $\boldsymbol{v}$ is the same discrete white Gaussian measurement noise as defined before. When a nonlinear measurement is available, Equation (2-22) is still used to determine the Kalman filter gain matrix, but the matrix $\boldsymbol{H}$ is defined by:

$$\boldsymbol{H}[t_i; \hat{\boldsymbol{x}}(t_i^-)] \triangleq \left. \frac{\partial \boldsymbol{h}[\boldsymbol{x}, t_i]}{\partial \boldsymbol{x}} \right|_{\boldsymbol{x} = \hat{\boldsymbol{x}}(t_i^-)} \tag{2-26}$$

The updated state vector becomes a function of the nonlinear residual, $[z(t_i) - \boldsymbol{h}[\hat{\mathbf{x}}(t_i^-), t_i)]$ and Equation (2-23) is modified to become:

$$\hat{\boldsymbol{x}}(t_i^+) = \hat{\boldsymbol{x}}(t_i^-) + \boldsymbol{K}(t_i)\left\{z(t_i) - \boldsymbol{h}[\hat{\boldsymbol{x}}(t_i^-), t_i]\right\} \tag{2-27}$$

2.2.3 Multiple Model Adaptive Algorithm. The optimality of the state estimator is dependent upon complete knowledge of the parameters that define the best model for system dynamics, output relations, and statistical description of uncertainties [22]. For Kalman filter tracking applications, maximum performance is achieved when the parameters of the filter dynamics model match the parameters of the target being tracked. Often, the parameters are known only with some uncertainty and may exhibit time-varying characteristics (such as in the case of maneuvering targets with changing acceleration levels). Thus, there is a need to devise a method that produces optimum state estimates despite the incomplete *a priori* knowledge of parameter values, and provides the estimates in an adaptive, on-line fashion. The MMAF satisfies these requirements [22].

To implement the MMAF algorithm, it becomes necessary to discretize the parameter space by the judicious choice of discrete values that are representatively dispersed throughout the continuous range of possible values. For the tracking problem at hand, a target can display K different discrete sets of pogo conditions corresponding to one of K discrete combinations of pogo oscillation frequency and magnitude. As previously shown in Figure 1.2, a Kalman filter is then designed for each choice of parameter value, resulting in a bank of K separate elemental filters.

Let $\boldsymbol{a}$ denote the vector of uncertain parameters in a given linear state model for a dynamic system. A system model would be represented by the following first-order, stochastic differential equation:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{F}(\boldsymbol{a})\boldsymbol{x}(t) + \boldsymbol{B}(\boldsymbol{a})\boldsymbol{u}(t) + \boldsymbol{G}(\boldsymbol{a})\boldsymbol{w}(t) \tag{2-28}$$

with noise corrupted, discrete-time measurements given by:

$$\boldsymbol{z}(t_i) = \boldsymbol{H}(\boldsymbol{a})\boldsymbol{x}(t_i) + \boldsymbol{v}(t_i) \tag{2-29}$$

where

$\boldsymbol{x}(t)$ = n-dimensional system state vector

$\boldsymbol{u}(t)$ = r-dimensional deterministic control vector

$\boldsymbol{w}(t)$ = s-dimensional white, Gaussian, zero-mean noise vector process of strength $\boldsymbol{Q}(\boldsymbol{a})$

$\boldsymbol{z}(t_i)$ = m-dimensional measurement vector

$\boldsymbol{v}(t_i)$ = m-dimensional discrete-time white, Gaussian, zero-mean noise vector process of covariance $\boldsymbol{R}(\boldsymbol{a})$

$\boldsymbol{F}(\boldsymbol{a})$ = n x n system plant matrix

$\boldsymbol{B}(\boldsymbol{a})$ = n x r input distribution matrix

$\boldsymbol{G}(\boldsymbol{a})$ = n x s noise distribution matrix

$\boldsymbol{H}(\boldsymbol{a})$ = m x n matrix relating measurement to states

The parameter vector, $\boldsymbol{a}$, is discretized into a set of K finite vector values, $\boldsymbol{a}_1$, $\boldsymbol{a}_2$,...$\boldsymbol{a}_K$, and associated with each $\boldsymbol{a}_k$ is a different system model of the form given by Equations (2-28) and (2-29). Each elemental Kalman filter, tuned for a specific $\boldsymbol{a}_k$, produces a state estimate which is weighed appropriately using the hypothesis conditional probability $p_k(t_i)$ to produce the state estimate $\hat{\boldsymbol{x}}_{mmaf}(t_i)$ as a probabilistically weighted sum, as shown in Figure 1.2 on page 1-9, where:

$$p_k(t_i) = \frac{f_{z(t_i)|a,Z(t_{i-1})}(\boldsymbol{z}_i|\boldsymbol{a}_k,\boldsymbol{Z}_{i-1})\,p_k(t_{i-1})}{\sum_{j=1}^{K} f_{z(t_i)|a,Z(t_{i-1})}(\boldsymbol{z}_i|\boldsymbol{a}_j,\boldsymbol{Z}_{i-1})\,p_j(t_{i-1})} \qquad (2\text{-}30)$$

$$f_{z(t_i)|a,Z(t_{i-1})}(\boldsymbol{z}_i|\boldsymbol{a}_k,\boldsymbol{Z}_{i-1}) = \frac{exp\{\cdot\}}{(2\pi)^{m/2}|\boldsymbol{A}_k(t_i)|^{1/2}}$$

$$\{\cdot\} = \left\{-\frac{1}{2}\boldsymbol{r}_k^T(t_i)\boldsymbol{A}_k^{-1}(t_i)\boldsymbol{r}_k(t_i)\right\} \qquad (2\text{-}31)$$

with

$\boldsymbol{A}_k(t_i)$ = kth filter's computed residual covariance
= $\boldsymbol{H}_k(t_i)\boldsymbol{P}_k(t_i^-)\boldsymbol{H}_k^T(t_i) + \boldsymbol{R}_k(t_i)$
$\boldsymbol{r}_k(t_i)$ = kth filter's residual
= $[\ \boldsymbol{z}(t_i) - \boldsymbol{H}_k(t_i)\hat{\boldsymbol{x}}_k(t_i^-)\ ]$
$\boldsymbol{a}_k$ = parameter value assumed in the kth filter
$\boldsymbol{P}_k(t_i^-)$ = kth filter's computed state error covariance before incorporating the measurement at time t_i
$\boldsymbol{Z}(t_{i-1})$ = measurement history up to time t_{i-1}

The residual of the kth elemental Kalman filter, that best matches the current pogo conditions associated with the parameter value $\boldsymbol{a}_k$, is expected to be smaller than the residuals of the other mismatched filters, so that the exponential term in Equation (2-31) is smallest for the kth elemental filter. Therefore, the hypothesis conditional probability given by Equation (2-30) with index corresponding to the "correct" filter will then be the largest among the conditional probabilities, thus assigning the most weight to the "correct" state estimate. This algorithm performs well if each elemental filter is optimally tuned for best performance for specific pogo conditions, causing its residual to be distinguishable from those of the mismatched filters. It is also important not to add excessive amounts of pseudonoise to compensate for model inadequacies, as is often done in conservative tuning of single Kalman filters, since this tends to mask the distinction between good and bad models [21]. If the quadratic forms within the exponentials of Equation (2-31) are consistently of the same magnitude, then Equation (2-30) will result in the growth of the p_k associated with the filter with the smallest value of $|A_k|$. The values of $|A_k|$ are independent not only of the residuals, but also of the "correctness" of the K models, and so the result would be totally erroneous [22]. Therefore, the scalar denominator of the exponential in Equation (2-31) might be removed in the final implementation of the algorithm.

The output of the MMAF algorithm is the probability-weighted average of the elemental filter's estimates given by:

$$\hat{x}_{mmaf}(t_i^+) = \sum_{k=1}^{K} p_k(t_i)\hat{x}_k(t_i^+) \qquad (2\text{-}32)$$

The conditional covariance matrix for the MMAF is computed as:

$$P_{mmaf}(t_1^+) = \sum_{k=1}^{K} p_k(t_i)[P_k(t_i^+) + \hat{y}_k(t_i^+)\hat{y}_k^T(t_i^+)] \qquad (2\text{-}33)$$

where

$\hat{y}_k(t_i^+) = \hat{x}_k(t_i^+) - \hat{x}_{mmaf}(t_i^+)$

p_k = kth filter's conditional hypothesis probability

$P_k(t_i^+)$ = kth filter's state error covariance matrix after incorporating the measurement at time t_i

Since the values of $p_k(t_i)$ and $\hat{x}_{mmaf}(t_i^+)$ depend upon the discrete measurements taken through time t_i, $P_{mmaf}(t_i^+)$ cannot be precomputed as in the case for the elemental filters. However, Equation (2-30) need not be computed for the on-line filter algorithm.

The calculated probabilities of Equation (2-30) should be modified by an artificial lower bound [17,22,37]. This lower bound will prevent a mismatched filter's hypothesis conditional probability from converging to (essentially) zero. If a filter's p_k should reach zero, it will remain zero for all time, as can be seen from the iterative nature of Equation (2-30). This effectively removes that filter from the bank and degrades the responsiveness of the MMAF to future changes of the parameter values. If some future pogo condition matched the model for which the p_k was locked onto zero, that elemental filter's estimate would not be appropriately weighted and the MMAF estimate would be in error. In previous work, Tobin [47] established a lower bound of .001 for $p_k(t_i)$.

2.3 Simulation Space

Simulation of the tracking scenario, which encompasses the target trajectory, the FLIR sensor operation, and the low-energy laser illumination of the missile hardbody and the generation of the speckle return and Doppler measurements, is performed on a digital computer. A 3-dimensional "simulation space" is generated wherein a target plume is propagated along a realistic trajectory. Several coordinate frames in the simulation space provide the means of mathematically projecting the target plume's infrared image and velocity vector onto the two-dimensional FLIR image plane [13,16,37]. In addition, these frames are utilized to project a representation of the hardbody center-of-mass, as well as to define the start and orientation of the low-energy laser scan for generating speckle and Doppler measurements [8,9]. This section describes the different coordinate frames of the simulation space and covers the process of pointing the FLIR sensor at the target during tracking. The coordinate frame descriptions and figures in this section are repeated from Herrera [14].

2.3.1 Coordinate Frames. As shown in Figure 2.1, three primary coordinate frames are defined in the simulation space: a system inertial reference frame, a target reference frame, and an α-β-r reference frame. Each of these reference frames is described in the following paragraphs.

2.3.1.1 Inertial Reference Frame. The inertial reference frame is a North-Up-East (NUE) frame wherein the target flight trajectory occurs.

Origin: location of the FLIR sensor
Axes: e_x - due north, tangent to the earth's surface, defines zero azimuth
e_y - inertial "up" with respect to flat earth approximation
e_z - vector completing right-hand coordinate set, defines 90° azimuth
Note: The azimuth angle (α) is measured eastward from e_x. The elevation angle (β) is measured "up" from the horizontal plane defined by e_x and e_z.

2.3.1.2 Target Plume Reference Frame. This frame is located at the target plume with one of its unit vectors co-linear with the target's velocity vector.

Origin: plume intensity centroid
Axes: e_v - along the true velocity vector
e_{pv} - out the right side of the target, orthogonal to both e_v and the LOS vector
e_{ppv} - vector completing the right-hand coordinate set

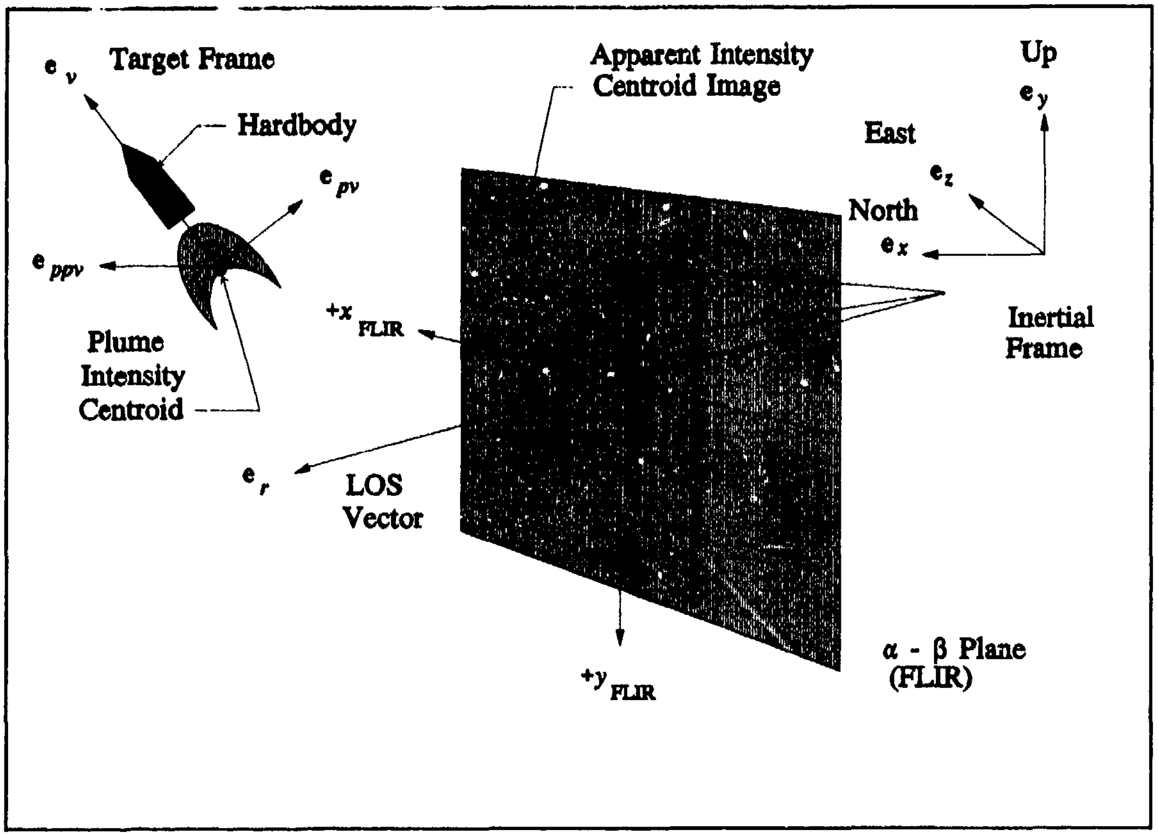


Figure 2.1 Three Primary Coordinate Frames in Simulation Space

Note: $_v$ - along the velocity vector
$_{pv}$ - perpendicular to the velocity vector
$_{ppv}$ - perpendicular to both $_v$ and $_{pv}$

2.3.1.3 α - β - r Reference Frame. The α-β-*r* reference frame is defined by the azimuth angle α′ and the elevation angle β′ measured with respect to the FLIR line-of-sight (LOS) vector e_r. The true azimuth α and the true elevation β are referenced from true north and the horizon. This frame is used to project the target's position and velocity onto the FLIR plane.

Origin: center of FLIR FOV
Axes: e_r, coincident with the true sensor-to-target LOS vector

e_α, and e_β define a plane perpendicular to e_r, rotated from inertial e_x and e_y by the azimuth angle (α) and elevation angle (β)

There are three special coordinate frames associated with the α-β-r reference frame: the α-β (FLIR) plane, the absolute α-β-r reference frame, and the trans-FLIR plane.

2.3.1.4 α - β (FLIR Image) Plane. The FLIR plane is used to obtain the measurements of the target plume position and is the reference frame for the geometrically derived velocity vector components of the target's intensity centroid. The FLIR plane is defined by the e_α and e_β unit vectors, with the LOS vector (orthogonal to the FLIR plane) representing the pointing orientation of the FLIR sensor, and the high and low-energy laser. Note the orientation of the $+y_{FLIR}$ axis in Figure 2.1, which allows the LOS vector to be positive towards the target when it is considered the third member of a right-handed set of coordinates as defined by the unit vectors e_β - e_α - e_r.

Due to the large distance to the target (approximately 2,000 kilometers), small angle approximations are invoked, allowing the "pseudo" azimuth and elevation angles, α' and β', to be linearly proportional to the x and y cartesian coordinates in the FLIR plane. The x and y coordinates are measured in pixels (a pixel of linear length corresponds to 15 μradians of arc) and will provide a means of evaluating the performance of the Kalman filter associated with tracking the intensity centroid of the target.

2.3.1.5 Absolute α-β-r Reference Frame. The absolute α-β-r reference frame is fixed in inertial space at the *initial* α-β-r coordinates of the target. This coordinate system defines the initial pointing direction of the FLIR LOS vector e_r, and is also used to define the true and filter estimated target positions and velocity components on the FLIR plane.

2.3.1.6 Trans-FLIR Plane. This plane is defined as the result of translating the center of the FLIR FOV to the true center-of-mass of the missile hardbody. The frame is used to determine the x_{FLIR} and y_{FLIR} coordinate errors of the hardbody center-of-mass filter's estimates, for performance analysis purposes.

2.3.1.7 ALT/ACT Plane. This plane, shown in Figure 2.2, is a rotation of the trans-FLIR plane by the true orientation angle θ_t, formed by the target trajectory with respect to the FLIR coordinate plane. It is used to determine the along-track and across-track components of the tracking error mean and covariance of the hardbody center-of-mass estimates [9]. This frame was not used in this thesis.

2.3.2 FLIR Image Plane. All dynamic events associated with the target plume intensity "pattern" or "function," and the active illumination of the missile hardbody in 3-dimensional inertial space are projected onto the 2-dimensional FLIR image plane. The measurements generated as a result of IR detection by the FLIR sensor are provided to the enhanced correlator algorithm, which produces "pseudo-measurements" to the Kalman filter to update its state estimates. For the missile hardbody, low-energy laser-generated measurements of the offset distance relative to the plume intensity centroid are geometrically projected onto the FLIR image plane. Thus, the FLIR image plane is the realm in which the performance of the Kalman filter is evaluated. Also note that it is a natural plane for such evaluation of a laser weapon, since pointing angle errors are critical and range is not. This section introduces the FLIR Field-Of-View (FOV) "tracking window," and discusses the construction and projection of the target models.

2.3.2.1 FLIR Field-Of-View. The FLIR FOV, shown in Figure 2.3, consists of an 8 x 8 pixel sub-array (in the FLIR sensor 300 x 500 pixel array) which provides sensed information as

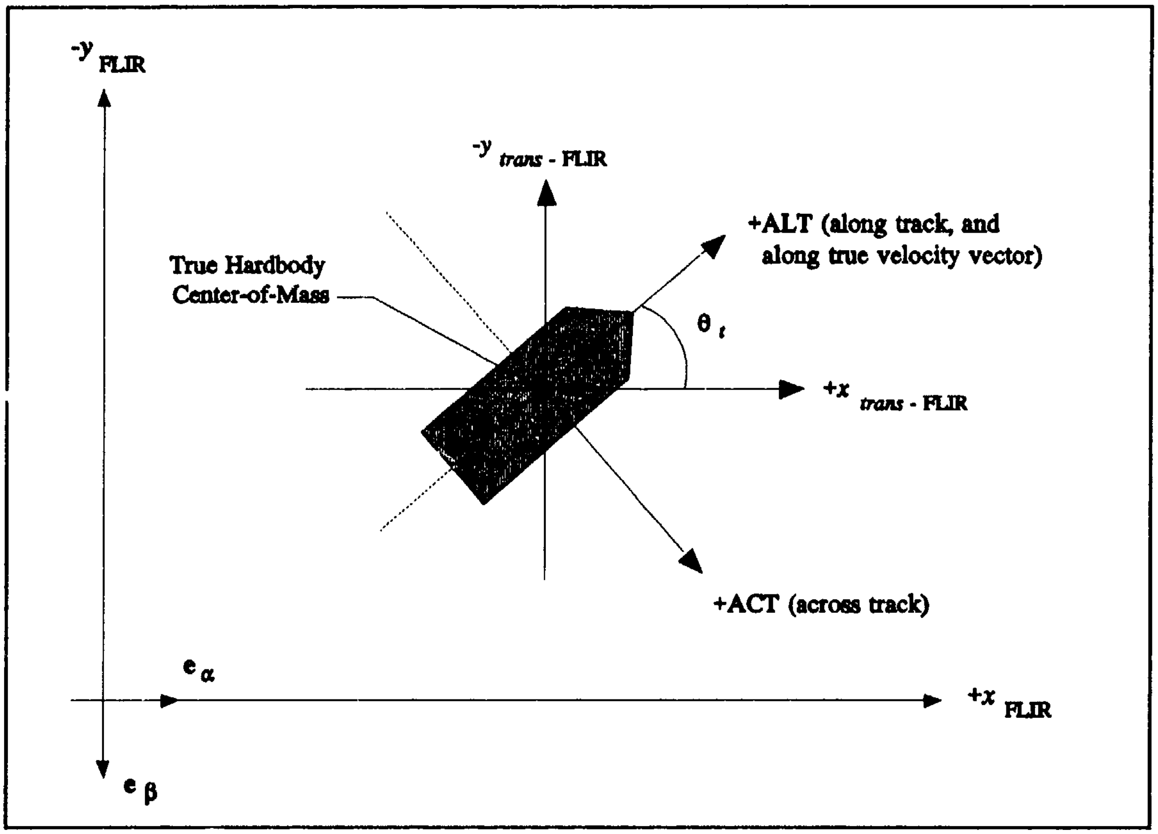

-y FLIR
-y trans - FLIR
+ALT (along track, and along true velocity vector)
True Hardbody Center-of-Mass
θ t
+x trans - FLIR
+ACT (across track)
e α
+x FLIR
e β

Figure 2.2 FLIR Plane, Trans-FLIR Plane, and ALT-ACT Plane

a function of the varying intensity of the plume IR image and the background and internal FLIR noise. Based upon this information, the position estimates from the nine-state Kalman filter serve to center the centroid of the plume IR image in the FOV. Since the low-energy laser is boresighted with the FOV, the filter position and velocity estimates of the intensity centroid define the origin and orientation of the laser scan to "paint" the hardbody. The errors of the filter's estimates of position, velocity, and centroid/center-of-mass offset, are expressed in units of "pixels" or "pixels"/second. These errors become meaningful through a pixel proportionality constant, k_p equal to 15 μradians/pixel [14,40]. With this constant, 1 pixel corresponds to approximately 30 meters for a range of 2,000 kilometers.

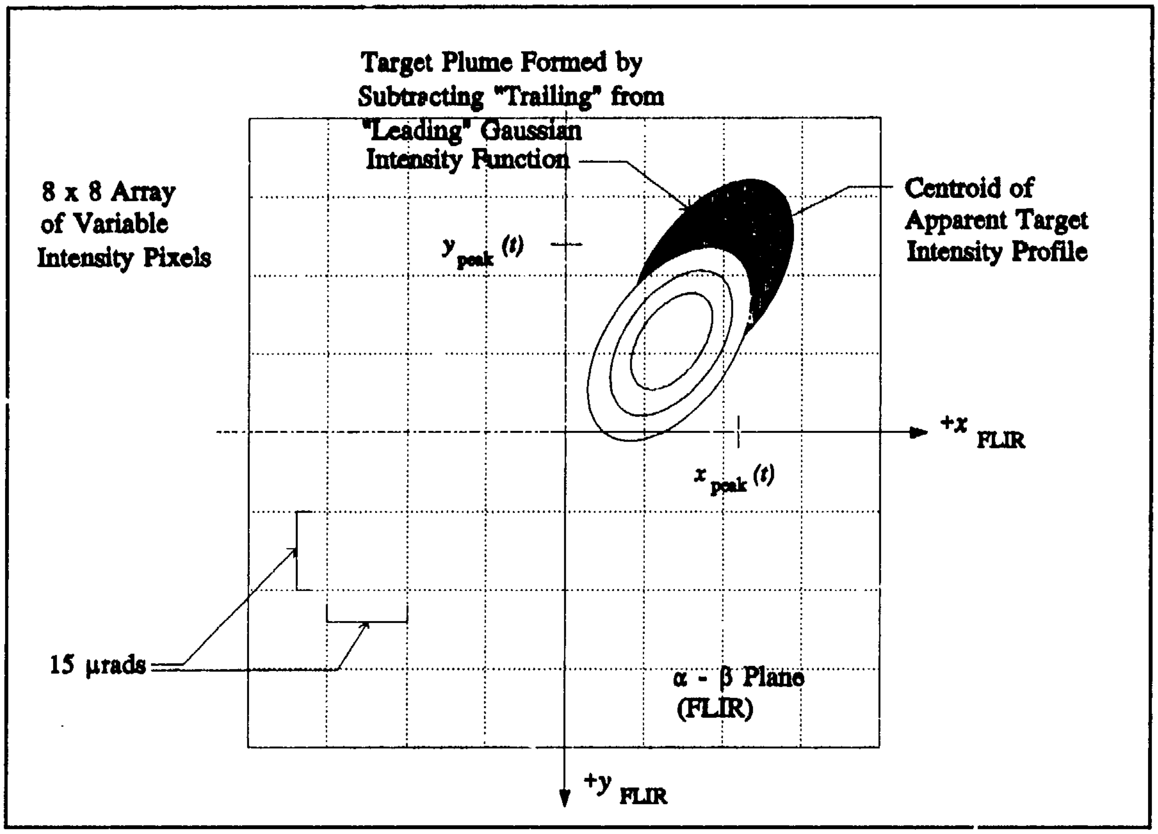


Figure 2.3 Target Plume Image in 8 x 8 FLIR Field-of-View (FOV)

2.3.2.2 Target Models on the FLIR Plane. The difference of two Gaussian intensity functions creates a planform that models the hotspot of the plume target on the FLIR plane [40], as shown in Figure 2.3. The "trailing" function is subtracted from the "leading" function to construct a suitable approximation of empirically observed plume intensity profiles. If the computed result is negative at some point, that negative value is replaced by zero. The missile hardbody is not sensed by the FLIR sensor. However, it is geometrically projected onto the FLIR plane as a rectangle, located an offset distance from the plume centroid along the target's velocity vector. Since the FLIR sensor can only detect the IR intensity shape function of the plume, the remainder of this discussion emphasizes the intensity centroid model. More about the hardbody model will be presented in Chapter III.

2.3.2.3 Target Plume Model on the FLIR Plane. The radiated energy from each intensity function is represented as a bivariate Gaussian distribution with elliptical constant intensity contours. Each of the two bivariate Gaussian intensity functions is given by [40]:

$$I\ [x,y,x_{peak}(t),y_{peak}(t)] = I_{max}\exp[-0.5(\Delta x\,\Delta y)\boldsymbol{P}^{-1}(\Delta x\,\Delta y)^{T}] \qquad (2\text{-}34)$$

where

Δx = $(x - x_{peak})\cos\theta_t + (y - y_{peak})\sin\theta_t$, measured along the along-track (ALT) axis of Figure 2.2

Δy = $(y - y_{peak})\cos\theta_t - (x - x_{peak})\sin\theta_t$, measured along the across-track (ACT) axis of Figure 2.2

θ_t = true target orientation angle between the projection of the velocity vector onto the FLIR plane and the x-axis in the FLIR plane; see Figure 3.2

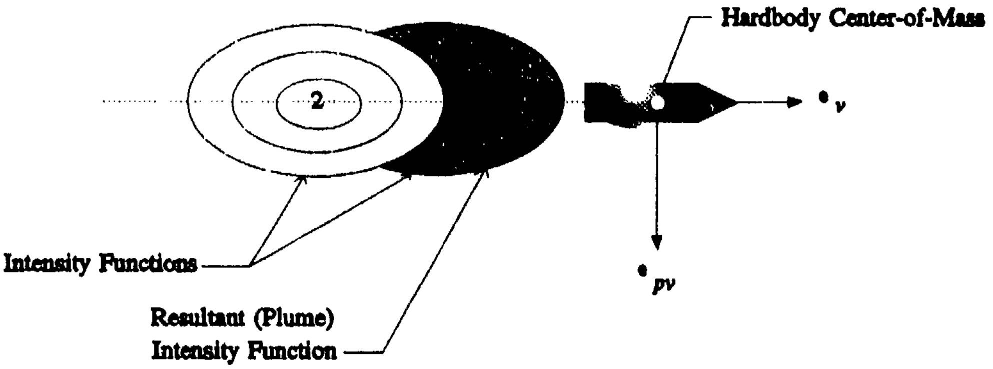
x, y = coordinate axes on the α - β plane

x_{peak}, y_{peak} = peak intensity coordinates of the single Gaussian intensity function

I_{max} = maximum intensity function

$\boldsymbol{P}$ = 2 x 2 target dispersion matrix with eigenvalues (σ_v^2 and σ_{pv}^2) that define the dispersion of the elliptical constant intensity contours, and eigenvectors that define the orientation of the principle axes of these ellipses

Figure 2.4 illustrates the spatial relationship between the two intensity functions along the target $\mathbf{e}_v$ axis. The displacement values are based on the assumption that the dispersion of the exhaust plume in the $\mathbf{e}_{pv}$ direction (normal to both $\mathbf{e}_v$ and the LOS vector) is approximately 20 *times* the diameter of the missile [40]. With the dimensions of the hardbody chosen as 40 meters long and 3 meters in diameter, the centroid of the first intensity function is located 65 meters behind the



Gaussian Intensity Function	Displacement of Centroid along e_v	Displacement of Centroid along e_{pv}	Displacement of Centroid along e_{ppv}
1	- 65 meters	0 meters	0 meters
2	- 110 meters	0 meters	0 meters

Figure 2.4 Spatial Relationship of Target Plume Gaussian Intensity Functions

hardbody center-of-mass. The placement of the first centroid simulates the composite centroid of the exhaust plume being close to the missile exhaust nozzle, whereas the position of the second centroid enables one to simulate different plume shapes. The second, "trailing" centroid is arbitrarily located 110 meters from the center-of-mass and the defined spatial relationship remains fixed in the target frame during the simulation. Any external forces acting on the missile other than thrust and gravity are assumed negligible, which thus yields an assumed zero sideslip angle as well as zero angle of attack. These assumptions allow the semi-major axes of the elliptical constant-intensity contours to be aligned with the projection of the target's velocity vector onto the FLIR image plane, and provides a simplified simulation geometry while retaining the essential features of the trajectory simulation.

2.3.2.4 Target Plume Projection onto the FLIR Plane. As the target plume is propagated through inertial space, the output of the FLIR pixels is simulated by projecting the two intensity functions onto the FLIR plane. The geometry of the projection is shown in Figure 2.5. The "reference target image" is oriented on the FLIR plane to correspond to the largest apparent planform (i.e., with its velocity vector orthogonal to the LOS vector) at a given initial reference range, r_o. As seen in Figure 2.6, the target intensity image is defined by the dispersion along the principle axes of the two Gaussian intensity functions, given by:

$$\sigma_{pv} = \sigma_{pvo}\left(\frac{r_o}{r}\right) \tag{2-35}$$

$$\begin{aligned} \sigma_v &= \left(\frac{r_o}{r}\right)\left[\sigma_{pvo} + (\sigma_{vo} - \sigma_{pvo})\cos\gamma\right] \\ &= \sigma_{pv}\left[1 + \frac{v_{\perp LOS}}{v}(AR - 1)\right] \end{aligned} \tag{2-36}$$

where

$\sigma_{vo}, \sigma_{pvo}$ = the initial dispersions of the target intensity functions along $\mathbf{e}_v$ and $\mathbf{e}_{pv}$ in the target frame of the reference image

σ_v, σ_{pv} = the current dispersions of the target image

r_o = initial sensor-to-target range of the reference image

r = current sensor-to-target range

$\mathbf{v}$ = initial velocity vector of the target

v = magnitude of $\mathbf{v}$

$\mathbf{v}_{\perp LOS}$ = projection of $\mathbf{v}$ onto the α - β plane (FLIR); i.e., the component of $\mathbf{v}$ perpendicular to the LOS vector

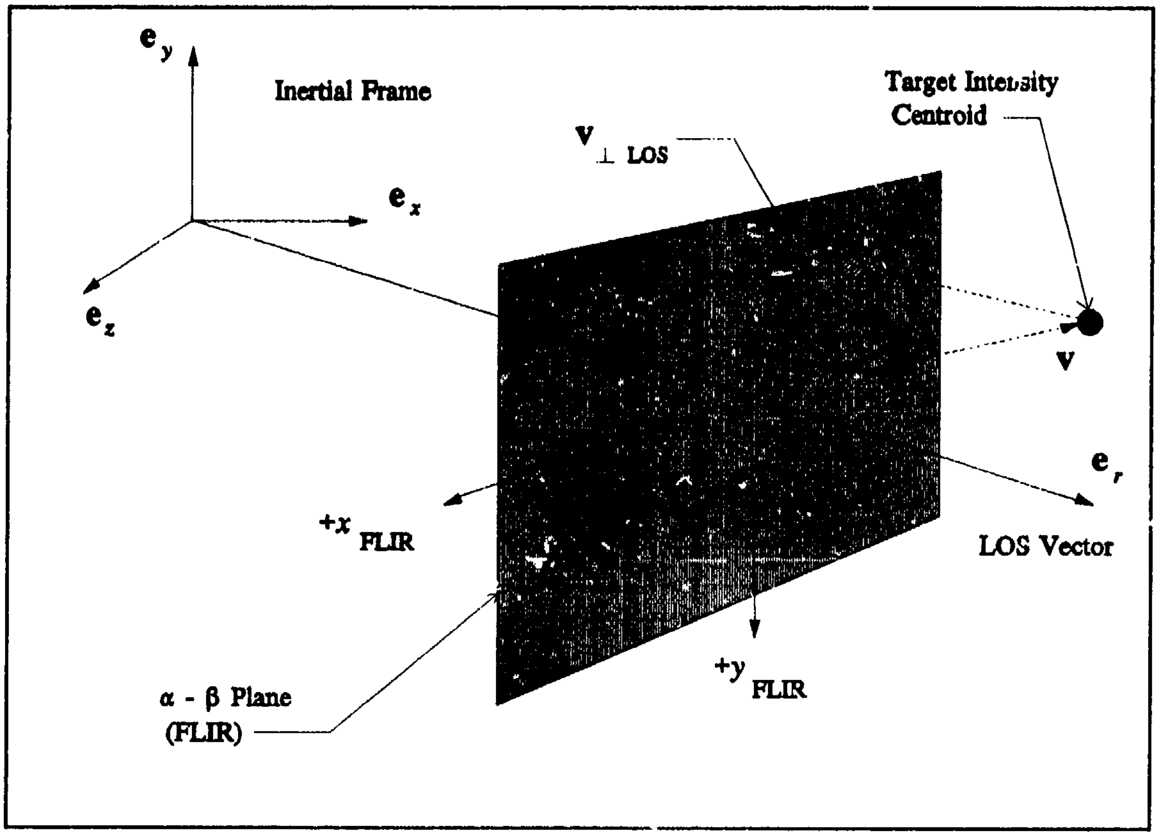


Figure 2.5 Target Plume Intensity Centroid Projection Geometry

$v_{\perp LOS}$ = magnitude of $\mathbf{v}_{\perp LOS}$:

$$v_{\perp LOS} = \sqrt{\dot{\alpha}^2 + \dot{\beta}^2} \tag{2-37}$$

γ = target aspect angle between v and the α - β plane (FLIR)

θ = angle between $\mathbf{v}_{\perp LOS}$ and $+x_{FLIR}$

AR = σ_{vo}/σ_{pvo}: aspect ratio of the reference image

Referring back to Figure 2.4, the location of each intensity function, or "hotspot," is initialized as a displacement from the hardbody center-of-mass. The intensity functions are oriented in the FLIR plane via the true target orientation angle θ_t. The relative positions of the

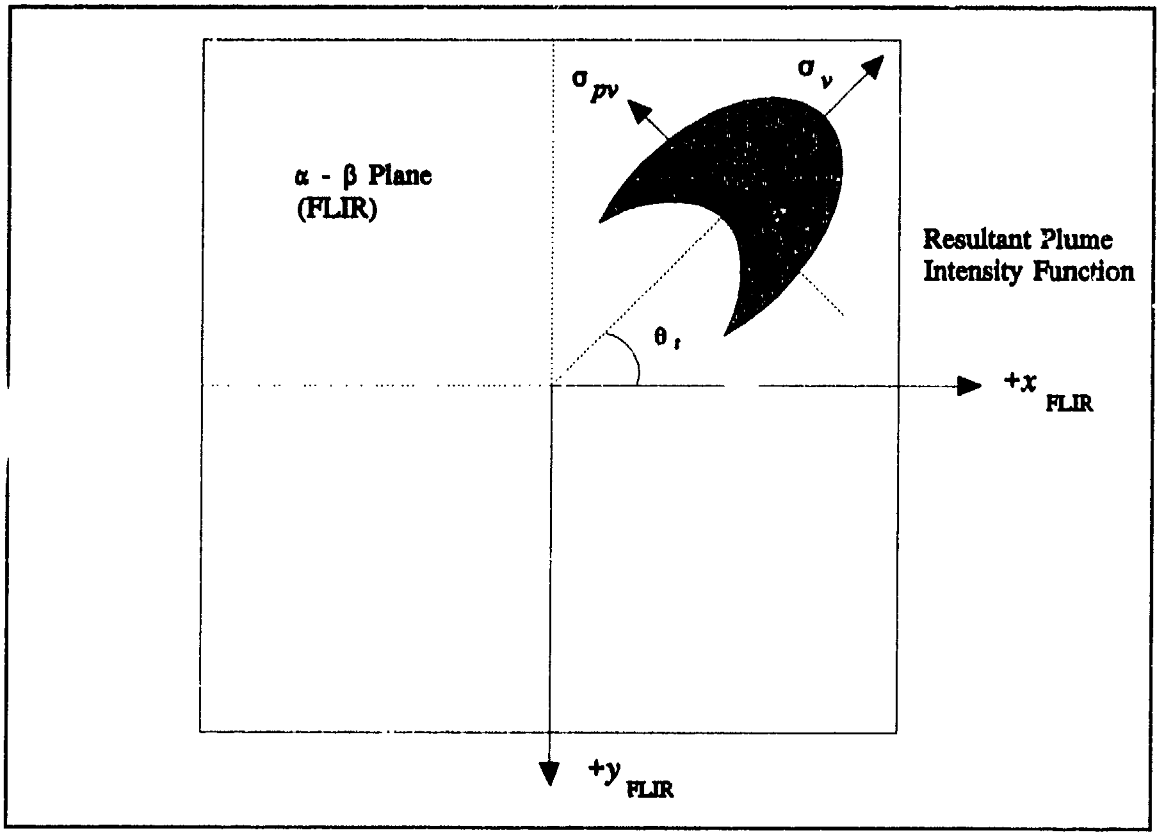


Figure 2.6 Intensity Centroid Dispersion Axes in FLIR Plane

two intensity functions in the FLIR plane vary in response to the change in target aspect angle γ (of Figure 2.5) while the spatial relationship of the hotspots remains the same in the three-dimensional target frame. If the plume pogo forcing input is applied, the hotspots do not remain fixed in the target frame, causing the composite image centroid to oscillate along the velocity vector and produce additional perturbations to the hotspot image in the FLIR plane, as will be described in the truth and filter model sections in Chapters III and IV respectively.

2.3.2.5 Target Plume Velocity Projection onto the FLIR Plane. The general discrete-time equation that models the target dynamics is given by:

$$\mathbf{x}(t_{i+1}) = \Phi(t_{i+1}, t_i)\mathbf{x}(t_i) + \mathbf{B}_d(t_i)\mathbf{u}(t_i) + \mathbf{G}_d(t_i)\mathbf{w}_d(t_i) \tag{2-38}$$

where

$\Phi(t_{i+1}, t_i)$ = the system state transition matrix
$\boldsymbol{x}(t_i)$ = discrete-time vector of states of interest
$\boldsymbol{B}_d(t_i)$ = discrete-time control input matrix
$\boldsymbol{u}(t_i)$ = discrete-time deterministic control input vector
$\boldsymbol{G}_d(t_i)$ = discrete-time driving noise input matrix
$\boldsymbol{w}_d(t_i)$ = discrete-time, zero-mean, white Gaussian noise process with independent components and covariance $\boldsymbol{Q}_d$

Based on the geometry shown previously in Figure 2.5, the projection of the target's inertial velocity vector onto the FLIR image plane is the deterministic input vector given by [13]:

$$\boldsymbol{u}_{td}(t_i) = \left[\dot{\alpha}'(t_i)\ \dot{\beta}'(t_i)\right]^T \tag{2-39}$$

where

$\boldsymbol{u}_{td}\, t_i)$ = true target deterministic input vector
$\dot{\alpha}'(t_i)$ = target azimuth rate in the FLIR plane
$\dot{\beta}'(t_i)$ = target elevation rate in the FLIR plane

As seen in the inertial frame diagrams of Figure 2.7, the azimuth angle can be defined as:

$$\alpha(t) = \arctan\left[\frac{z(t)}{x(t)}\right] \tag{2-40}$$

Taking the time derivative of Equation (2-40) and noting that the sensor-to-target range is large so that $\dot{\alpha}'(t_i) = \dot{\alpha}(t_i)$, the azimuth velocity in the FLIR plane is given by:

$$\dot{\alpha}'(t) = \dot{\alpha}(t) = \frac{x(t)v_z(t) - z(t)v_x(t)}{x^2(t) + z^2(t)} \tag{2-41}$$

where

v_x, v_z = components of the target's inertial velocity in the e_x and e_z directions

Similarly, the elevation velocity in the FLIR plane is given by:

$$\dot{\beta}'(t) = \dot{\beta}(t) = \frac{r_h(t)v_y(t) - y(t)\dot{r}_h(t)}{r^2(t)} \tag{2-42}$$

where

v_y = component of the target's inertial velocity in the e_y direction

r_h = horizontal projection of the sensor-to-target range, with its time derivative expressed as:

$$\dot{r}_h(t) = \frac{x(t)v_z(t) + z(t)v_x(t)}{r_h(t)} \tag{2-43}$$

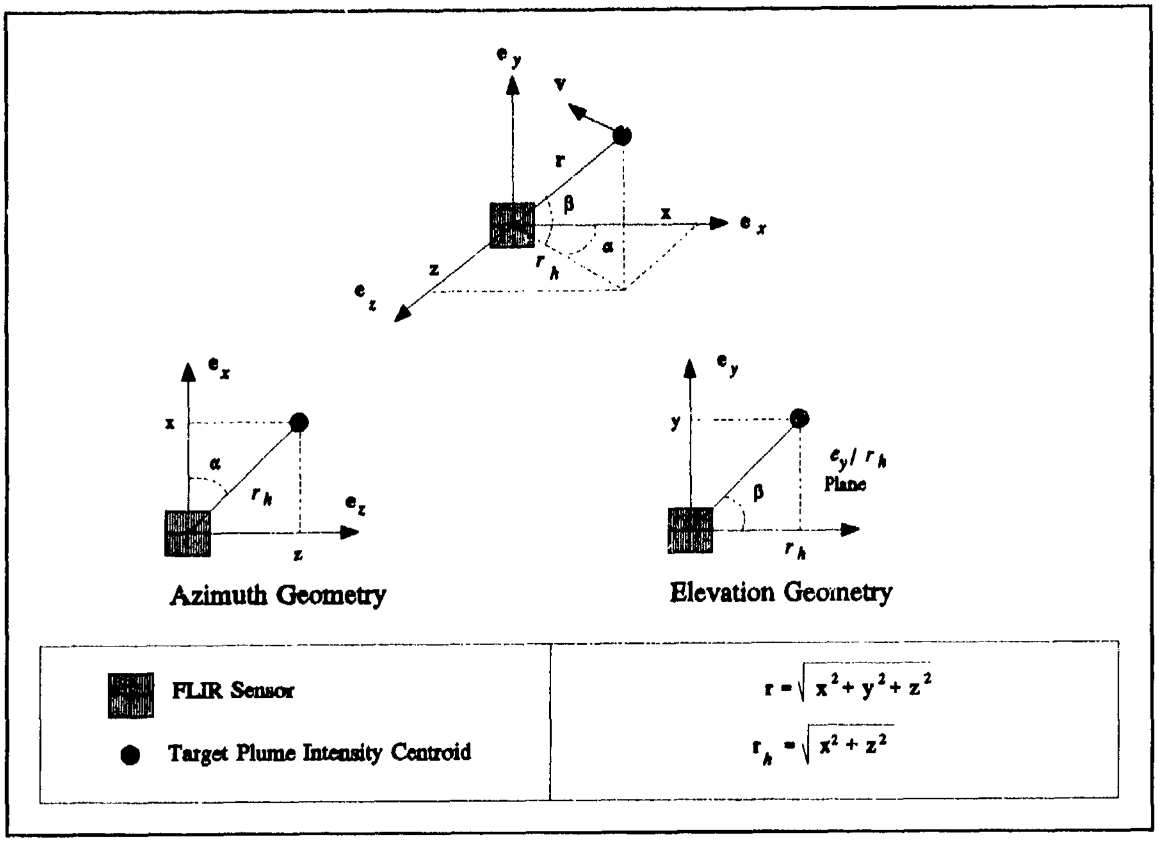


Figure 2.7 Inertial Velocity FLIR Plane Projection Geometry

2.3.3 FLIR Sensor Pointing Controller. The Kalman filter's propagated estimates of the intensity centroid's position dictate the necessary change in azimuth and elevation that the FLIR sensor should undergo over the next sample period to center the hotspot on the FLIR FOV plane at the next measurement sample time. Ideally, these positional estimates are fed as commands to a pointing controller that physically implements the directional changes within one sample period (1/60 sec). The original sample period used in the recent past for the benign trajectory scenarios has been 1/30 second. For this thesis, the sample period was cut in half for two major reasons. First, it was hypothesized that the reduced sample period might allow for closer inspection of the difficulties surrounding estimation of the atmospheric jitter states. If the reduced sample period does not aid this investigation, it should at least reduce the differences in errors at the beginning and end of the propagation period. The second reason that the sample period was reduced to 1/60 second was to maintain conformity with the current hardware and software being developed at the Phillips Laboratory. The most recent programs utilizing the FLIR and LEL measurements for tracking and pointing a high energy laser are using the 1/60 second sample period [4].

The activation and execution of the commands to the FLIR for directional changes will not be perfect due to the lag dynamics inherent in the controller, and the resultant mis-positioning of the hotspot may be interpreted by the filter as target motion, causing inaccurate estimates of future states. Whether or not to include the controller lag dynamics in the simulation was the subject of a previous thesis [37]. It was found that the apparent target motion caused by the lag dynamics are interpreted by the filter as atmospheric jitter (as a result of the choice of tuning parameters for the filter), implying a degree of robustness on the part of the filter to track a target. Moveover, the degradation in tracking performance due to the dynamic lag was found not to be of primary importance. Thus, the controller is modeled as lag-free in this research.

2.3.4 Target Scenarios. Two different scenarios were utilized for simulations in this research. The first scenario was employed for both the stochastic observability tests and the MSOFE Monte Carlo runs. The tracker was given the proper initial conditions of the target (to similate a handoff from another tracker) and was to track a missile possessing the characteristics and following the profile of a Minuteman II [4,5]. Over the ten second simulation period, a nominal intercept range of 200 km and negligible acceleration were assumed. In the MSOFE simulations, the truth model was programmed to maintain a constant velocity equal to the average value determined over the simulation period [5]. As with the reduced sample period, the reduced range (from 2000 Km to 200 Km) was derived from intercept scenarios recently developed for use at Phillips laboratory. The intent is eventually to revise the scenario modeled in the AFIT software and compare filter performance under the new conditions.

The second scenario, used for the Monte Carlo simulations in the AFIT software and referenced throughout this thesis, was the same as used in previous theses. An Atlas type missile is simulated traveling at a range of 2000 km from the platform with a constant velocity. The original intent of the runs was to complete a few simulations with the old parameters and then eventually to update the conditions to match the first scenario. This was not done for two major reasons. The first reason was that a direct comparison of the new filter's performance with previous filters was desired and the second was that the new filter including pogo states was never satisfactorily tuned to elicit its maximum performance potential. A complete description of all initial conditions and flight parameters of the truth model is contained with the truth model description in Chapter III.

2.4 Summary

This chapter presented the theoretical background required to understand the research explained in the rest of the thesis. It briefly developed the linear Kalman filter for both continuous and discrete-time processing, and it also presented the extended Kalman filter modification to the update equations to allow for nonlinear measurement models to be used. The mathematical models presented in the Kalman filter section will relate directly to the truth and filter models described in Chapters III and IV respectively. Section three of this chapter contained a complete description of the various coordinate frames used for the simulations. The different coordinate frames were required in order to simulate the scenario realistically in inertial 3-space and to transform the missile trajectory into coordinate systems useful for sensor and computer processing. The third section also outlined the inter-relationships of the coordinate frames. Finally, this chapter briefly summarized the pointing controller for the HEL and the two target scenarios exploited in the thesis.

III. Truth Model

3.1 Introduction

In order to analyze the tracking algorithms of interest successfully, they must be subjected to as many of the dynamic variables that would be encountered in the real world as possible. The truth model is a mathematical model derived for simulation purposes which embodies all of pertinent states necessary to create a realistic environment in which the filter will operate [21]. In theory, an absolute truth model could be of infinite dimension. In practice, only the dominant characteristics that will affect the system of interest need be modeled. The truth model should be developed through thorough testing and analysis of data collected in the real world from the measuring instruments. Shaping filters should then be developed and compared to the empirical data to validate the model.

This chapter will present the truth filter models used for development and testing of the elemental linear Kalman filters in the AFIT simulation. It will also cover the models used to generate measurements for the filter updates and the detailed parameters used for the truth model simulation (initial conditions, etc). Since the AFIT truth model has remained virtually unchanged since the previous thesis, the portions discussing the model were taken from Chapter IV of Herrera's thesis [14] with minor alterations.

The models used for MSOFE [6] were reduced-order versions of the model used in the AFIT software and described in Section 3.2, that incorporated a subset of the major characteristics desired for the observability studies. They were also implemented as continuous-time dynamics/discrete-time measurement models in MSOFE as opposed to the discrete-time dynamics equations employed in the AFIT software. The truth models and filter models used for MSOFE were identical except for the different tuning values that adjusted noise levels in the filter. The reader should review the next section for an understanding of the entire truth model and refer to Chapter IV for the specific models used for MSOFE analyses.

3.2 Truth Dynamics Model

The essential conditions which must be generated in the AFIT truth model include the underlying dynamic states of the target, jitter from atmospheric disturbances, bending/vibration of the host platform, and measurement noises in the sensors. As previously noted in Chapter I, the truth model was originally developed when only FLIR measurements were available. As a result, the truth dynamics model defines the target in terms of the image intensity centroid, *not* the missile center-of-mass. The true location of the hardbody is determined by adding a constant centroid to center-of-mass offset (87.5 m) to the equilibrium point about which the centroid oscillates. The system describing the target image includes the following fourteen states [14,17,31,40]:

2 target dynamic states
6 atmospheric states
4 mechanical bending states
2 pogo oscillation states

The dynamics of the image intensity centroid are represented as positional changes in the FLIR plane, with the centroid components x_c and y_c being measured in pixels from the center of the FOV in the x and y FLIR plane directions. Referring to Figure 3.1, the position of the target image centroid at any one time is given by:

$$x_c = x_t + x_a + x_b + x_p\cos\theta_t \tag{3-1}$$

$$y_c = y_t + y_a + y_b - x_p\sin\theta_t \tag{3-2}$$

where

x_c, y_c = target image intensity centroid coordinates

x_t, y_t = coordinate deviation due to target dynamics

x_a, y_a = coordinate deviation due to atmospheric jitter

x_b, y_b = coordinate deviation due to bending/vibration of optical hardware

x_p = coordinate deviation due to pogo oscillations along the velocity vector direction

θ_t = true target orientation angle

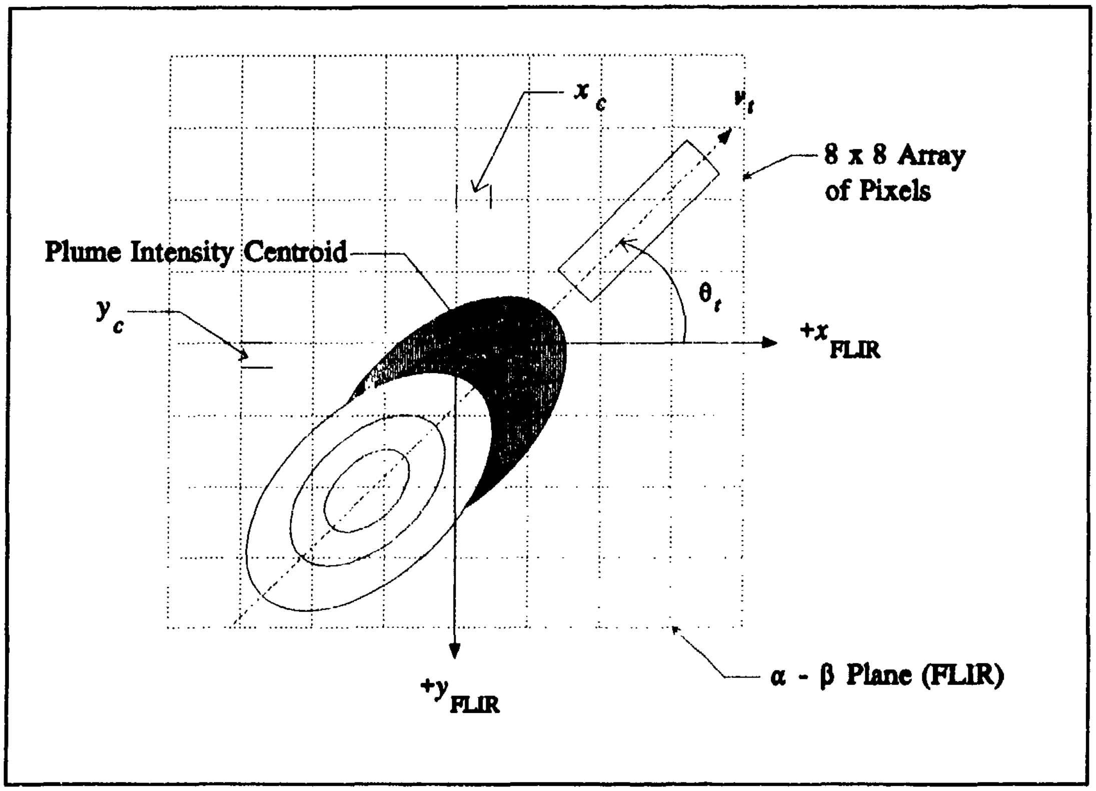


Figure 3.1 Plume Intensity Function Position on FLIR Image Plane

Note the minus sign before the resolved pogo component in Equation (3-2) due to the coordinate definition of the FLIR coordinate frame. The states x_a, x_b, x_t, x_p, y_a, y_b, and y_t comprise the output states which are extracted from an overall state model in the form of fourteen coupled scalar stochastic differential equations. The states x_t and y_t are each modeled by first-order differential equations; x_b, y_b, and x_p are each modeled by second-order differential equations; and x_a and y_a are each modeled with third-order differential equations. These differential equations, when in state-space format, comprise the dynamics portion of the FLIR tracker truth model.

The 14-state model state vector is described by a first-order, stochastic differential equation given by (note the use of the capitol T subscript to denote reference to the truth model, as opposed to a lower case t to reference variables associated with target dynamics):

$$\dot{\mathbf{x}}_T(t) = \mathbf{F}_T \mathbf{x}_T(t) + \mathbf{B}_T \mathbf{u}_T(t) + \mathbf{G}_T \mathbf{w}_T(t) \tag{3-3}$$

where

$\mathbf{F}_T$ = 14 x 14 time-invariant truth model plant matrix

$\mathbf{x}_T(t)$ = 14-dimensional truth model state vector

$\mathbf{B}_T$ = 14 x 2 time-invariant truth model control distribution matrix

$\mathbf{u}_T(t)$ = 2-dimensional deterministic input vector

$\mathbf{G}_T$ = 14 x 14 noise distribution matrix ($\mathbf{G}_T = \mathbf{I}$)

$\mathbf{w}_T(t)$ = 14-dimensional, white Gaussian noise process with mean and covariance kernel statistics:

$$\begin{aligned} E\{\mathbf{w}_T(t)\} &= 0 \\ E\{\mathbf{w}_T(t)\mathbf{w}_T^T(t + \tau)\} &= \mathbf{Q}_T \delta(t) \end{aligned} \tag{3-4}$$

To simulate the target dynamics model on a digital computer, the following equivalent discrete-time solution to Equation (3-3) is given by:

$$\boldsymbol{x}_T(t_{i+1}) = \Phi_T(t_{i+1}, t_i)\boldsymbol{x}_T(t_i) + \boldsymbol{B}_{Td}\boldsymbol{u}_{Td}(t_i) + \boldsymbol{G}_{Td}\boldsymbol{w}_{Td}(t_i) \tag{3-5}$$

where the state transition matrix $\Phi_T(t,t_i)$ is the solution to the differential equation:

$$\frac{d\Phi_T(t, t_i)}{dt} = \boldsymbol{F}_T\Phi_T(t, t_i) \tag{3-6}$$

with the initial condition: $\Phi_T(t_i, t_i) = \boldsymbol{I}$, (note that, for constant $\boldsymbol{F}_T$, $\Phi_T(t,t_i)$ can be expressed as $\Phi_T(t-t_i)$) and

$\boldsymbol{x}_T(t_i)$ = 14-dimensional discrete-time truth model state vector

$\boldsymbol{B}_{Td}$ = 14 x 2 discrete-time truth model control distribution matrix

$\boldsymbol{u}_{Td}(t_i)$ = 2-dimensional discrete-time input vector

$\boldsymbol{G}_{Td}$ = 14 x 14 discrete-time noise distribution matrix, ($\boldsymbol{G}_{Td} = \boldsymbol{I}$)

$\boldsymbol{w}_{Td}(t_i)$ = 12-dimensional discrete time, white Gaussian noise process with mean and covariance statistics:

$$E\{\boldsymbol{w}_{Td}(t_i)\} = \boldsymbol{0} \tag{3-7}$$

$$E\{\boldsymbol{w}_{Td}(t_i)\boldsymbol{w}_{Td}^T(t_i)\} = \boldsymbol{Q}_{Td} = \int_{t_i}^{t_{i+1}} \Phi_T(t_{i+1} - \tau)G_T Q_T G_T^T \Phi_T^T(t_{i+1} - \tau)\,d\tau \tag{3-8}$$

where $\boldsymbol{Q}_T$ is defined in Equation (3-4). The discrete-time input distribution matrix $\boldsymbol{B}_{Td}$ is defined as:

$$\boldsymbol{B}_{Td} = \int_{t_i}^{t_{i+1}} \Phi_T(t_{i+1} - \tau)\boldsymbol{B}_T\,d\tau \tag{3-9}$$

Note that this computation assumes $\boldsymbol{u}_T(t)$ is constant over each sample period: $\boldsymbol{u}_T(t) = \boldsymbol{u}_{Td}(t_i)$ for all $t \in [t_i, t_{i+1})$. This input simulates a true constant velocity trajectory for the missile.

The fourteen states of the discrete-time truth model are defined in the *x* and *y* coordinate axes of the FLIR plane as:

x_{FLIR}	y_{FLIR}
1 target state	1 target state
3 atmospheric states	3 atmospheric states
2 bending/vibration states	2 bending/vibration states

2 plume pogo states (position and velocity

where the plume pogo states are in neither the x_{FLIR} nor y_{FLIR} direction but defined along the velocity vector. These states are augmented into the truth model state vector:

$$x_T = \begin{bmatrix} x_t \\ \cdots\cdots \\ x_a \\ \cdots\cdots \\ x_b \\ \cdots\cdots \\ x_p \end{bmatrix} \tag{3-10}$$

where

x_t = 2-dimensional target dynamics state vector
x_a = 6-dimensional atmospheric state vector
x_b = 4-dimensional bending/vibration state vector
x_p = 2-dimensional plume pogo state vector

The 14 x 14 discrete-time truth model state transition matrix Φ_T is given by:

$$\Phi_T = \begin{bmatrix} \Phi_t & \vdots & 0 & \vdots & 0 & \vdots & 0 \\ \cdots\cdots & & \cdots\cdots & & \cdots\cdots & & \cdots\cdots \\ 0 & \vdots & \Phi_a & \vdots & 0 & \vdots & 0 \\ \cdots\cdots & & \cdots\cdots & & \cdots\cdots & & \cdots\cdots \\ 0 & \vdots & 0 & \vdots & \Phi_b & \vdots & 0 \\ \cdots\cdots & & \cdots\cdots & & \cdots\cdots & & \cdots\cdots \\ 0 & \vdots & 0 & \vdots & 0 & \vdots & \Phi_p \end{bmatrix} \tag{3-11}$$

where the partitions correspond to the dimensionality of the states defined above. The 14 x 2 discrete-time truth model distribution matrix B_{Td} is given by:

$$B_{T_d} = \begin{bmatrix} B_{td} \\ \cdots\cdots \\ 0 \end{bmatrix} \tag{3-12}$$

where B_{T_d} is a 2 x 2 discrete-time control distribution matrix. The 14-dimensional discrete-time truth model white Gaussian noise process w_{T_d} is given by:

$$w_{T_d} = \begin{bmatrix} 0 \\ \cdots\cdots \\ w_{ad} \\ \cdots\cdots \\ w_{bd} \\ \cdots\cdots \\ w_{pd} \end{bmatrix} \tag{3-13}$$

where

$w_{ad}(t_i)$ = 6-dimensional discrete-time, white Gaussian noise related to atmospheric jitter states

$w_{bd}(t_i)$ = 4-dimensional discrete-time, white Gaussian noise related to bending states

$w_{pd}(t_i)$ = 2-dimensional discrete-time, white Gaussian noise related to plume pogo states

The block diagonal form of Equation (3-5), as seen in Equations (3-10) through (3-13), allows the models for target dynamics, atmospheric jitter, bending/vibration, and plume pogo to be presented separately. The following sections discuss each of the discrete state models which form the stochastic discrete-time truth model.

3.2.1 Target Dynamics State Description. As depicted in Figure 3.2, the α-β plane (FLIR image plane) is coincident with the FLIR sensor FOV, and perpendicular to the LOS vector e_r. In the simulation, the 3-dimensional target dynamics are projected onto the FLIR image plane, and the position components of the target's intensity centroid are obtained from the azimuth and elevation displacement angles (α' and β', respectively). Since the target distance is simulated

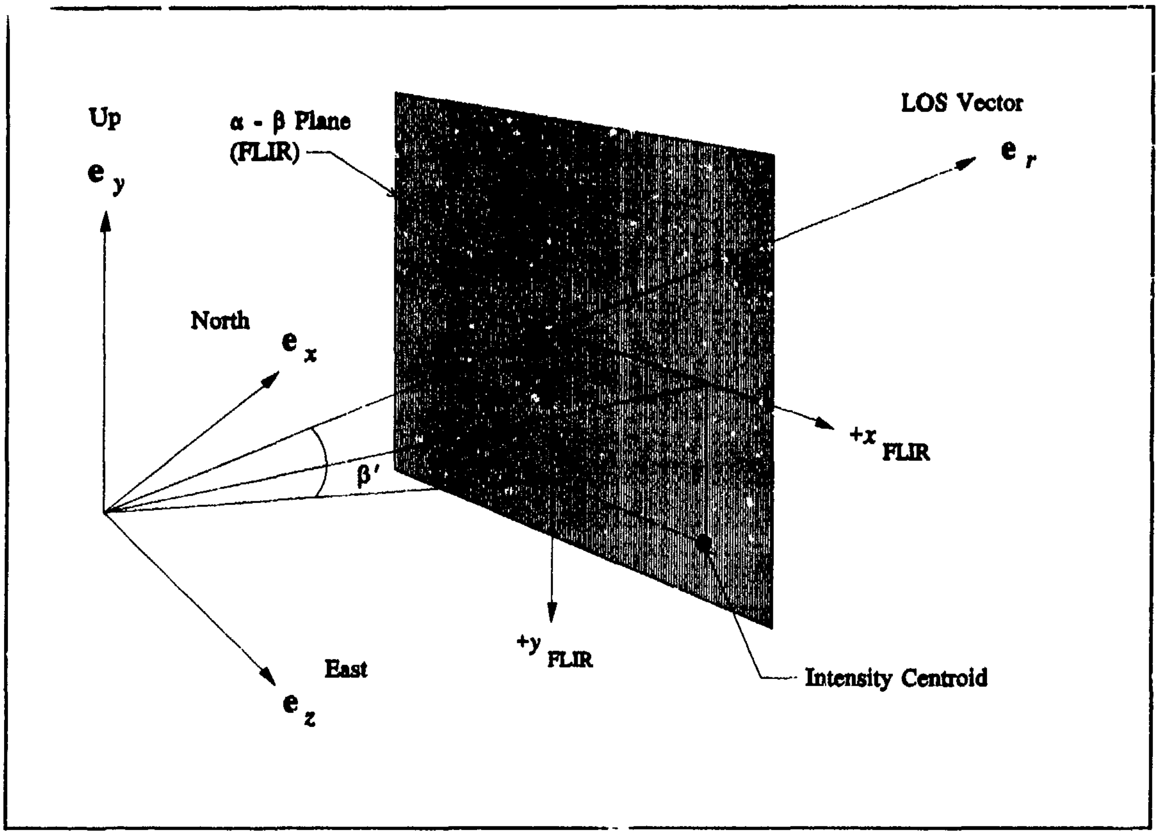


Figure 3.2 Target Centroid Image on α-β Plane with "Pseudo" Angles

as 2,000 kilometers, small angle approximations are used for measuring the angle displacements in the cartesian coordinate system of the FLIR image plane. These "pseudo" angles, α' and β', are referenced from the current LOS vector and measured in microradians. Note that the unusual orientation of the $+y_{FLIR}$ axis in Figure 3.2 allows the positive z axis to be in the positive e_r direction (by the right-hand rule).

The linear translational coordinates, x_c and y_c of Equations (3-1) and (3-2), locate the target intensity function on the FLIR plane and are measured in pixels of displacement from the center of the FLIR FOV. The angular and linear measurements are related by the pixel proportionality constant k_p, which is the angular FOV of a single pixel. Presently, the value of k_p is approximately 15 microradians per pixel for long range targets [8,40].

3.2.1.1 AFIT Discrete-Time Target Dynamics. The derivation of the state space model of the target dynamics assumes that the azimuth and elevation rates ($\dot{\alpha}'$ and $\dot{\beta}'$, respectively) remain essentially constant over each sample period Δt. Then the discrete-time target dynamics model is:

$$x_t(t_{i+1}) = x_t(t_i) + \frac{(\dot{\alpha}')(\Delta t)}{k_p} \tag{3-14}$$

$$y_t(t_{i+1}) = y_t(t_i) - \frac{(\dot{\beta}')(\Delta t)}{k_p} \tag{3-15}$$

Arranging these equations in state space form yields:

$$\mathbf{x}_t(t_{i+1}) = \Phi_t(t_{i+1},t_i)\mathbf{x}_t(t_i) + \mathbf{B}_t\mathbf{u}_t(t_i) \tag{3-16}$$

$$\begin{bmatrix} x_t(t_{i+1}) \\ \\ y_t(t_{i+1}) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_t(t_i) \\ \\ y_t(t_i) \end{bmatrix} + \begin{bmatrix} \frac{\Delta t}{k_p} & 0 \\ \\ 0 & -\frac{\Delta t}{k_p} \end{bmatrix} \begin{bmatrix} \dot{\alpha}'(t_i) \\ \\ \dot{\beta}'(t_i) \end{bmatrix} \tag{3-17}$$

where

$\dot{\alpha}'(t_i)$ = $d\alpha'/dt$, measured in microradians/second and constant over the time interval Δt

$\dot{\beta}'(t_i)$ = $d\beta'/dt$, measured in microradians/second and constant over the time interval Δt

Δt = sample time interval, $t_{i+1} - t_i$ (1/60 seconds)

k_p = pixel proportionality constant (15 microradians/pixel)

Using these relationships in block form of the overall truth model, by inspection of Equation (3-11), the upper left block is:

$$\Phi_t = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{3-18}$$

and the upper block of Equation (3-12) is:

$$B_{td} = \begin{bmatrix} \dfrac{\Delta t}{k_p} & 0 \\ 0 & -\dfrac{\Delta t}{k_p} \end{bmatrix} \tag{3-19}$$

and the input vector in Equation (3-5) is given by:

$$u_{Td} = \begin{bmatrix} \dot{\alpha}'(t_i) \\ \dot{\beta}'(t_i) \end{bmatrix} \tag{3-20}$$

The minus sign of the lower right term in Equation (3-19) is due to the difference in the y axis orientation between the inertial coordinate frame and the FLIR coordinate plane.

The two target dynamics states of Equation (3-10) are used to propagate the missile along its trajectory. The formulation of the truth model target dynamics states in deterministic state space form has two advantages. First, Equation (3-17) can be substituted back into Equation (3-5) to form a single augmented vector differential equation that defines the truth model. Second, the state space form allows the addition of white (or time-correlated) noise to Equation (3-17), if a stochastic, rather than a deterministic dynamics model, is desired.

3.2.1.2 MSOFE Continuous-Time Target Dynamics. As shown in Chapter IV, an equivalent continuous-time dynamics constant velocity model was simulated in MSOFE by letting the deterministic input vector of Equation (3-20) equal zero. Two velocity states (along the FLIR plane axes) were augmented to the state vector for a total of four states describing target dynamics. The desired constant velocity trajectory is achieved by designing the plant matrix, $\boldsymbol{F}_t$, in Equation (3-3) such that the target dynamics are modeled by $\dot{\boldsymbol{x}} = \boldsymbol{v}$ and $\dot{\boldsymbol{v}} = \boldsymbol{a} = \boldsymbol{0}$.

3.2.2 Atmospheric Jitter Model. The model for the translational displacement of the intensity function due to atmospheric disturbances, is based on a study by The Analytic Sciences Corporation [31]. Using power spectral density characteristics, the atmospheric jitter phenomenon in each FLIR plane axis direction can be modeled as the output of a third-order shaping filter driven by white Gaussian noise [31]. The Laplace domain representation of the shaping filter transfer function is given by:

$$\frac{x_a(s)}{w_a(s)} = \frac{K_a \omega_1 \omega_2^2}{(s + \omega_1)(s + \omega_2)^2} \tag{3-21}$$

x_a = output of shaping filter (x_{FLIR} direction)
w_a = zero-mean, scalar, unit-strength white Gaussian noise
K_a = gain, adjusted for desired atmospheric jitter rms value
ω_1 = break frequency, 14.14 radians/second
ω_2 = double-pole break frequency, 659.5 radians/second

The atmospheric jitter effects can be modeled similarly in the y_{FLIR} direction, where y_a would be the output of an identical shaping filter defined in Equation (3-21). The two shaping filters are assumed to be independent of each other and can thus be augmented to form a six-state model. The linear stochastic differential equation that describes the atmospheric jitter is given by:

$$\dot{\boldsymbol{x}}_a(t) = \boldsymbol{F}_a \boldsymbol{x}_a(t) + \boldsymbol{G}_a \boldsymbol{w}_a(t) \tag{3-22}$$

where

$\boldsymbol{F}_c$ = 6 x 6 time-invariant atmospheric jitter plant matrix
$\boldsymbol{x}_a(t)$ = 6-dimensional atmospheric jitter state vector

G_a = 6 x 2 noise distribution matrix

$w_a(t)$ = 2-dimensional, independent, zero mean white Gaussian noise with unit strength and independent components described as:

$$E\{w_a(t_i)\} = 0$$

$$E\{w_a(t_i)w_a^T(t_i)\} = Q_a\delta(\tau) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\delta(\tau) \tag{3-23}$$

The six atmospheric states in the state vector correspond to the low frequency pole and the higher frequency double pole in the x_{FLIR} and the y_{FLIR} directions. The atmospheric jitter plant matrix is defined in Jordan Canonical form as:

$$F_a = \begin{bmatrix} -\omega_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\omega_2 & 1 & 0 & 0 & 0 \\ 0 & 0 & -\omega_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\omega_1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\omega_2 & 1 \\ 0 & 0 & 0 & 0 & 0 & -\omega_2 \end{bmatrix} \tag{3-24}$$

The noise distribution matrix G_a is:

$$G_a = \begin{bmatrix} \frac{K_a\omega_1\omega_2^2}{(\omega_1 - \omega_2)^2} & 0 \\ -\frac{K_a\omega_1\omega_2^2}{(\omega_1 - \omega_2)^2} & 0 \\ \frac{K_a\omega_1\omega_2^2}{(\omega_1 - \omega_2)} & 0 \\ 0 & \frac{K_a\omega_1\omega_2^2}{(\omega_1 - \omega_2)^2} \\ 0 & -\frac{K_a\omega_1\omega_2^2}{(\omega_1 - \omega_2)^2} \\ 0 & \frac{K_a\omega_1\omega_2^2}{(\omega_1 - \omega_2)} \end{bmatrix} \tag{3-25}$$

The equivalent discrete-time model for Equation (3-22) is of the form:

$$x_a(t_{i+1}) = \Phi_a(t_{i+1}, t_i)x_a(t_i) + w_{da}(t_i) \tag{3-26}$$

The augmented six-state state transition matrix derived from the time-invariant plant matrix of Equation (3-24) is [31]:

$$\Phi_a(\Delta t) = \begin{bmatrix} \Phi_{a11} & 0 & 0 & 0 & 0 & 0 \\ 0 & \Phi_{a22} & \Phi_{a23} & 0 & 0 & 0 \\ 0 & 0 & \Phi_{a33} & 0 & 0 & 0 \\ 0 & 0 & 0 & \Phi_{a44} & 0 & 0 \\ 0 & 0 & 0 & 0 & \Phi_{a55} & \Phi_{a56} \\ 0 & 0 & 0 & 0 & 0 & \Phi_{a66} \end{bmatrix} \tag{3-27}$$

where

$$\Phi_{a11} = \Phi_{a44} = \exp(-\omega_1\Delta t)$$

$$\Phi_{a22} = \Phi_{a55} = \exp(-\omega_2\Delta t)$$

$$\Phi_{a23} = \Phi_{a56} = \Delta t\,\exp(-\omega_2\Delta t)$$

$$\Phi_{a33} = \Phi_{a66} = \exp(-\omega_2 \Delta t)$$

$$\Delta t = \text{sample time interval, } t_{i+1} - t_i$$

The 6-dimensional, zero-mean, discrete-time, white, Gaussian noise $w_{ad}(t_i)$ has statistics defined as:

$$E\{w_{ad}(t_i)\} = 0$$

$$E\{w_{ad}(t_i)w_{ad}^T(t_i)\} = Q_{ad} = \int_{t_i}^{t_{i+1}} \Phi_a(t_{i+1}-\tau) G_a Q_a G_a^T \Phi_a^T(t_{i+1}-\tau)\, d\tau \qquad (3\text{-}28)$$

The individual components of Q_{ad} are not included here due to their length and complexity; however the reader may refer to the software for a full description [30].

For the approximated two-state atmospheric jitter model, only a single-pole shaping filter is used in each direction to produce the approximated PSD. The state space equations are truncated from six to two states with only the first the break frequency, ω_1, used in each direction. The plant matrix in Equation (3-24) becomes a 2 x 2 with $-\omega_1$ as the diagonal terms and Equation (3-25) also becomes 2 x 2 with $K_a\omega_1$ on the diagonal.

3.2.3 Bending/Vibration Model. The mechanical bending states were added to the truth model to account for the vibrational effects in the FLIR data that occur when the sensor is mounted on a moving, non-rigid optical platform [17]. Based on tests at the AFWL (now Phillips Laboratory), it was concluded in previous research [17] that bending effects in both the x_{FLIR} and y_{FLIR} directions can be represented by a second order shaping filter, driven by white Gaussian noise. The Laplace domain transfer function for the bending model is:

$$\frac{x_b(s)}{w_b(s)} = \frac{K_b\omega_{nb}^2}{s^2 + 2\zeta_b\omega_{nb}s + \omega_{nb}^2} \qquad (3\text{-}29)$$

where

x_b = FLIR plane positional offset (x_{FLIR} direction) due to mechanical bending disturbance

w_b = zero-mean, unit strength, white Gaussian noise

K_b = gain adjustment to obtain desired rms bending output, ($K_b^2 = 5 \times 10^{-13} \text{rad}^4/\text{sec}^4$)

ζ_b = damping coefficient, equal to 0.15

ω_{nb} = undamped natural frequency for bending, ($\omega = \pi$ rad/sec)

The FLIR plane positional offset in the y_{FLIR} direction, y_b, is identically modeled with the shaping filter defined in Equation (3-29). The two shaping filters are assumed to be independent of each other and can thus be augmented to form a four-state model. The linear stochastic differential equation that describes the bending/vibration is given by:

$$\dot{\mathbf{x}}_b(t) = \mathbf{F}_b \mathbf{x}_b(t) + \mathbf{G}_b \mathbf{w}_b(t) \tag{3-30}$$

where

$\mathbf{F}_b$ = 4 x 4 time-invariant bending plant matrix

$\mathbf{x}_b(t)$ = 4-dimensional bending state vector

$\mathbf{G}_b$ = 4 x 2 noise distribution matrix

$\mathbf{w}_b(t)$ = 2-dimensional, white Gaussian noise process with unit strength components that are independent of each other:

$$E\{\mathbf{w}_b(t)\} = \mathbf{0}$$

$$E\{\mathbf{w}_b(t)\mathbf{w}_b^T(t+\tau)\} = \mathbf{Q}_b\delta(\tau) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\delta(\tau) \tag{3-31}$$

The bending/vibration plant matrix is defined as:

$$F_b = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\omega_{nb}^2 & -2\zeta_b\omega_{nb} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\omega_{nb}^2 & -2\zeta_b\omega_{nb} \end{bmatrix} \tag{3-32}$$

The noise distribution matrix G_b is:

$$G_b = \begin{bmatrix} 0 & 0 \\ \omega_{nb}^2 k_p & 0 \\ 0 & 0 \\ 0 & \omega_{nb}^2 k_p \end{bmatrix} \tag{3-33}$$

(Note that k_p is the pixel proportionality constant.) The equivalent discrete-time model for Equation (3-30) is of the form:

$$x_b(t_{i+1}) = \Phi_b(t_{i+1}, t_i) x_b(t_i) + w_{bd}(t_i) \tag{3-34}$$

where

$$\Phi_b(\Delta t) = \begin{bmatrix} \Phi_{b11} & \Phi_{b12} & 0 & 0 \\ \Phi_{b21} & \Phi_{b22} & 0 & 0 \\ 0 & 0 & \Phi_{b33} & \Phi_{b34} \\ 0 & 0 & \Phi_{b43} & \Phi_{b44} \end{bmatrix} \tag{3-35}$$

and

$\Phi_{b11} = \Phi_{b33} = \exp(-\sigma_b\Delta t)[\cos(\omega_b\Delta t) + (\sigma_b/\omega_b)\sin(\omega_b\Delta t)]$

$\Phi_{b12} = \Phi_{b34} = \exp(-\sigma_b\Delta t)[(1/\sigma_b)\sin(\omega_b\Delta t)]$

$\Phi_{b21} = \Phi_{b43} = \exp(-\sigma_b\Delta t)[-1 - (\sigma_b/\omega_b)^2 \sin(\omega_b\Delta t)]$

$\Phi_{b22} = \Phi_{b44} = \exp(-\sigma_b\Delta t)[\cos(\omega_b\Delta t) - (\sigma_b/\omega_b)\sin(\omega_b\Delta t)]$

Δt = sample time interval, $t_{i+1}-t_i$

σ_b = real part of the root of the characteristic equation in Equation (3-29), ($\sigma_b = 0.47124$ second^{-1})

ω_b = imaginary part of the root of the characteristic equation in Equation (3-29), (ω_b = 3.10605 radians/second)

The 4-dimensional, discrete-time, white Gaussian noise process vector $\mathbf{w}_{db}(t_i)$ has mean and covariance statistics:

$$E\{w_{bd}(t_i)\} = \mathbf{0}$$

$$E\{w_{bd}(t_i)w_{bd}^T(t_i)\} = Q_{bd} = \int_{t_i}^{t_{i+1}} \Phi_b(t_{i+1}-\tau)G_bQ_bG_b^T\Phi_b^T(t_{i+1}-\tau)\,d\tau \qquad (3\text{-}36)$$

3.2.4 Plume Pogo Model. To account for the oscillatory nature of a typical missile plume in the boost phase, a plume pogo model was developed [40]. A second-order Gauss-Markov model was generated using physical insight, and visual observation of the pogo phenomenon. The model allows for the study of the amplitude and frequency characteristics of the oscillatory nature of the plume, and of the effect upon tracking a missile using a Kalman filter.

The transfer function of the plume pogo model is described in the Laplace domain as:

$$\frac{x_p(s)}{w_p(s)} = \frac{K_p\omega_{np}^2}{s^2 + 2\zeta_p\omega_{np}s + \omega_{np}^2} \qquad (4\text{-}37)$$

where

x_p = plume pogo shaping filter output along the direction of the velocity vector

w_p = zero-mean, unit strength, white Gaussian noise

ζ_p = assumed damping coefficient, (ζ = 0.05)

ω_{np} = nominal undamped natural frequency for pogo; assumed range is 0.1 - 10 Hertz, with a nominal value of 1.0 Hertz

K_p = gain adjustment to obtain desired rms pogo amplitude determined by [40]:

$$K_p = 2\sigma_p\sqrt{\frac{\zeta_p}{\omega_{np}}} \tag{3-38}$$

where

σ_p = desired rms pogo along the velocity vector

The linear stochastic differential equation that describes the plume pogo is given in state space form as:

$$\dot{x}(t_i) = \begin{bmatrix} 0 & 1 \\ -\omega_{np}^2 & -2\zeta_p\omega_{np} \end{bmatrix} x_p(t) + \begin{bmatrix} 0 \\ K_p\omega_{np}^2 \end{bmatrix} w_p(t) \tag{3-39}$$

where

$x_p(t)$ = 2-dimensional pogo state vector composed of pogo position and velocity states

$w_p(t)$ = 1-dimensional zero-mean, white Gaussian noise with statistics:

$$\begin{aligned} E\{w_p(t)\} &= 0 \\ E\{w_p(t)w_p(t+\tau)\} &= Q_p\delta(t-\tau); \quad Q_p = 1 \end{aligned} \tag{3-40}$$

The equivalent discrete-time model for Equation (3-39) is of the form:

$$x_p(t_{i+1}) = \Phi_p(t_{i+1},t_i)x_p(t_i) + w_{pd}(t_i) \tag{3-41}$$

$$x_p(t_{i+1}) = \begin{bmatrix} \Phi_{p11}(\Delta t) & \Phi_{p12}(\Delta t) \\ \Phi_{p21}(\Delta t) & \Phi_{p22}(\Delta t) \end{bmatrix} x_p(t_i) + w_{pd}(t_i) \tag{3-42}$$

where

$$
\begin{aligned}
\Phi_{p11}(\Delta t) &= \frac{1}{\sqrt{1-\zeta_p^2}}\exp(-\zeta_p\omega_{np}\Delta t)\sin\left[\omega_{np}\sqrt{1-\zeta_p^2}\,\Delta t + \arctan\left(\frac{\sqrt{1-\zeta_p^2}}{\zeta_p}\right)\right] \\
\Phi_{p12}(\Delta t) &= \frac{1}{\omega_{np}\sqrt{1-\zeta_p^2}}\exp(-\zeta_p\omega_{np}\Delta t)\sin\left(\omega_{np}\sqrt{1-\zeta_p^2}\,\Delta t\right) \\
\Phi_{p21}(\Delta t) &= \frac{-\omega_{np}}{\sqrt{1-\zeta_p^2}}\exp(-\zeta_p\omega_{np}\Delta t)\sin\left(\omega_{np}\sqrt{1-\zeta_p^2}\,\Delta t\right) \\
\Phi_{p22}(\Delta t) &= \frac{1}{\sqrt{1-\zeta_p^2}}\exp(-\zeta_p\omega_{np}\Delta t)\sin\left[\omega_{np}\sqrt{1-\zeta_p^2}\,\Delta t + \arctan\left(\frac{\sqrt{1-\zeta_p^2}}{-\zeta_p}\right) + \pi\right]
\end{aligned}
\tag{3-43}
$$

The 2-dimensional, discrete time, white Gaussian noise process $w_{pd}(t_i)$ has mean and covariance statistics:

$$
\begin{aligned}
E\{w_{pd}(t_i)\} &= 0 \\
E\{w_{pd}(t_i)w_{pd}^T(t_i)\} = Q_{pd} &= \int_{t_i}^{t_{i+1}} \Phi_p(t_{i+1}-\tau)G_pQ_pG_p^T\Phi_p^T(t_{i+1}-\tau)\,d\tau
\end{aligned}
\tag{3-44}
$$

The individual components of Q_{pd} are not included here due to their length and complexity. The interested reader is referred to the software [30] for a full description.

The 2-dimensional pogo state vector defines the position of the plume image intensity centroid relative to the equilibrium point of oscillation, and its velocity component due to the pogo phenomenon along the longitudinal axis of the missile. For the simulation, it is assumed that the velocity vector lies coincident with the longitudinal axis of the hardbody. As shown in Figure 1.6, the plume oscillates about an equilibrium point also located on the longitudinal axis. This equilibrium point is defined by the initial positions of the two intensity functions in the target coordinate frame (to be discussed in Section 3.3.1), and remains at a constant distance from the hardbody center-of-mass throughout the simulation (the spatial relationship of the intensity functions can be seen in Figure 3.3 in Section 3.3.1). The crescent-shaped plume represents one of many equal-intensity contour lines of the actual plume. The angle of attack and sideslip angle of the missile are also assumed negligible, and have zero values for the simulation [40].

3.3 Measurement Models

A major factor in formulating an authentic simulation is creating realistic measurements for the Kalman filter state estimate updates. The measurements generated in a simulation are based on prior knowledge and/or physical insight into the characteristics of the sensor devices. This knowledge not only includes what the measurement properties themselves should be for the chosen scenario (i.e. dependent on the current state of the truth *and* filter model), but also any imperfections that would exist due to background noises in the environment and the measurement devices themselves. Two types of measurements are provided to the linear Kalman filter in this simulation.

The first type are the "pseudo-measurements" generated by the enhanced correlation algorithm from the raw FLIR data [41]. These two measurements describe the position of the intensity centroid after undergoing changes due to dynamics, atmospheric jitter, bending/vibration and pogo oscillations. In the truth model, the image intensity function is corrupted with two types of measurement noise before being processed by the enhanced correlation algorithm -- spatially correlated noise from the target background, and spatially uncorrelated noise to simulate internal FLIR sensor noises.

The second type of measurement fed to the filter is from the low energy laser (LEL). This measurement is obtained by scanning the LEL relative to the plume centroid and recording any backscatter that may (or may not) result from intercepting the hardbody. Since the Kalman filter may utilize the magnitude of speckle return, the Doppler frequency spectra attributes, or both, the generated LEL measurement must emulate all those properties. For this research, only the Doppler measurement model was used. The objective in this simulation was not to recreate the physical processes of the laser speckle (or Doppler) return itself, but to imitate and provide the expected quality of measurements based on those processes, the chosen scenario, and the various scan techniques employed. Noises to account for imperfections and device limitations are also modeled in the LEL measurements.

The following subsections describe how the measurements were generated for this simulation. A full description of the FLIR model is included as well as how LEL reflections from the plume and hardbody are modeled. In addition, a brief review of the Doppler effect is included to assist in understanding the section on how Doppler measurements are modeled. As with the truth dynamics model, the measurement model was not revised from the last thesis, so the following descriptions were transcribed from Herrera's thesis [14] with modifications made to ensure the information is germane to this thesis.

3.3.1 FLIR Model. The FLIR sensor model is composed of an 8 x 8 pixel array "tracking window" extracted from the total array of 300 x 500 pixels. The missile plume is projected onto the FLIR focal plane, with its characteristic crescent-shaped intensity function formed as the difference of two bivariate Gaussian intensity functions, as shown in Figure 3.3. This model depends upon knowledge of several parameters: the size of the major and minor axes of the elliptical contours of each bivariate Gaussian function, and the orientation of the principal axes in the FLIR image plane (the major axis of each ellipse points along the velocity vector in the

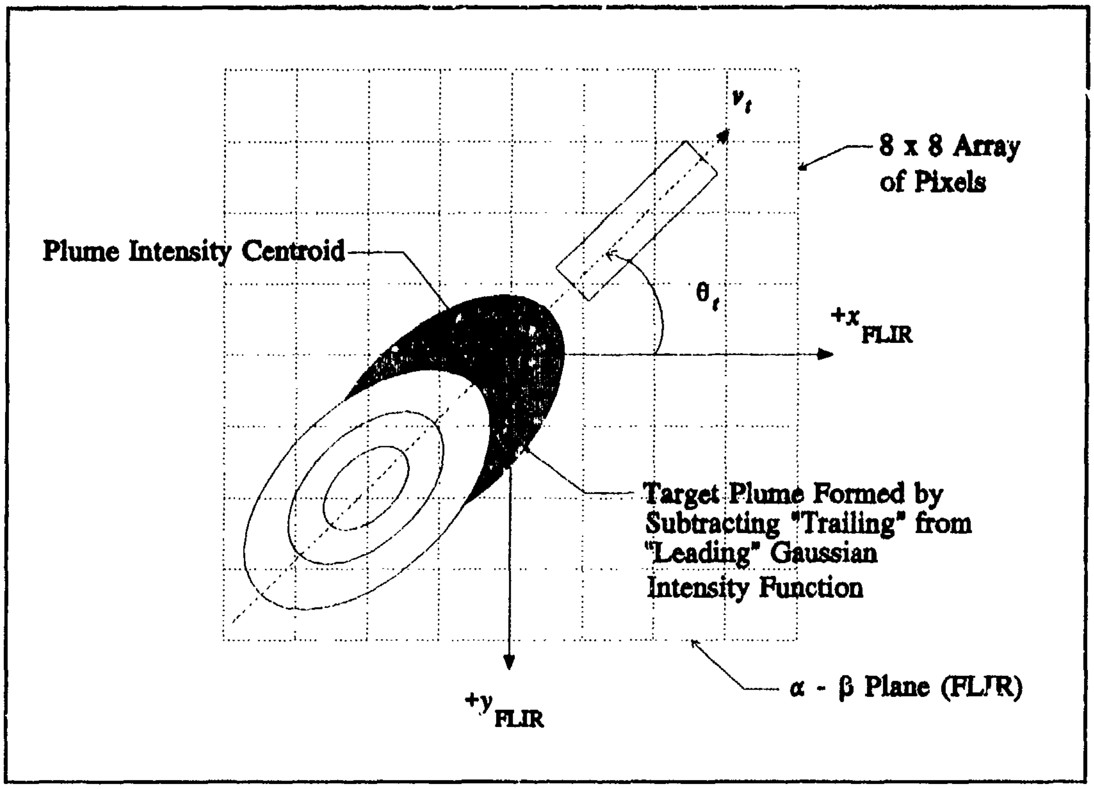


Figure 3.3 Composite Plume Intensity Function on FLIR Plane

FLIR plane). The target intensity function obtained from evaluating the resulting intensity function is corrupted by spatially correlated and temporally uncorrelated background noise and spatially and temporally uncorrelated internal FLIR noise, according to models of actual data taken from a FLIR sensor looking at various backgrounds [38].

For each pixel in the FLIR FOV (the 8 x 8 array "tracking window"), the target's intensity function, correlated background noise, and FLIR internal noise are added together to produce an intensity measurement. For the 8 rows and 8 columns of the FOV, the intensity measurement corresponding to the pixel in the j^{th} row and k^{th} column at sampling time t_i is given by:

$$
\begin{aligned}
z_{jk}(t_i) = \frac{1}{A_p}\int_{pixel_{jk}} \{ I_1[x,y,x_{peak1}(t_i),y_{peak1}(t_i) \\
- I_2[x,y,x_{peak2}(t_i),y_{peak2}(t_i)]\} dx\,dy \\
+ n_{jk}(t_i) + b_{jk}(t_i)
\end{aligned} \tag{3-45}
$$

where

$z_{jk}(t_i)$ = output of pixel in the j^{th} row and k^{th} column

A_p = area of one pixel

I_1, I_2 = intensity function of first and second Gaussian intensity function respectively of Figure 3.3

x, y = coordinates of any point within pixel jk

x_{peak1}, y_{peak1} = coordinates of maximum point of first Gaussian intensity function

x_{peak2}, y_{peak2} = coordinates of maximum point of second Gaussian intensity function

$n_{jk}(t_i)$ = effect of internal FLIR sensor noise on jk^{th} pixel

$b_{jk}(t_i)$ = effect of spatially correlated background noise on jk^{th} pixel

The sensor error, $n_{jk}(t_i)$, is the result of thermal noise and dark current in the IR detectors (pixels). This error is assumed to be both temporally and spatially uncorrelated [40].

The background noise, $b_{jk}(t_i)$, was observed in the FLIR data by AFWL personnel during a tracking operation [13]. It is represented as a spatially correlated noise with radial symmetry, with a correlation that decays exponentially. Harnly and Jensen [13] concluded that spatial correlation can be depicted as a correlation distance of approximately two pixels in the FLIR plane, and simulated this by maintaining non-zero correlation coefficients between each pixel and its two closest neighbors symmetrically in all directions. In that two-pixel distance, the correlation decays exponentially to one-tenth of its peak value.

The generation of spatially correlated white Gaussian noises is accomplished by allowing non-zero cross correlations between the measurement noises, $b_{ik}(t_i)$, associated with each of the 64 pixels from the 8 x 8 pixel FLIR FOV. The correlated measurement noise in Equation (3-45) is given as:

$\boldsymbol{b}(t_i)$ = 64-dimensional vector of spatially correlated noise with statistics:

$$\begin{aligned} E\{\boldsymbol{b}(t_i)\} &= \boldsymbol{0} \\ E\{\boldsymbol{b}(t_i)\boldsymbol{b}^T(t_j)\} &= \boldsymbol{R}\delta_{ij} \end{aligned} \qquad (3\text{-}46)$$

where $\boldsymbol{R}$ is a 64 x 64 measurement noise covariance matrix. This matrix describes the spatial correlation between pixels, and is given by [13]:

$$\boldsymbol{R} = \sigma_R^2 \begin{bmatrix} 1 & r_{1,2} & r_{1,3} & \cdots & r_{1,64} \\ r_{2,1} & 1 & r_{2,3} & \cdots & r_{2,64} \\ r_{3,1} & r_{3,2} & 1 & \cdots & r_{3,64} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ r_{64,1} & r_{64,2} & r_{64,3} & \cdots & 1 \end{bmatrix} \qquad (3\text{-}47)$$

where σ_R^2 is the variance of each scalar noise and the correlation coefficients $r_{j,k}$ are evaluated to reflect the radially symmetric, exponentially decaying pattern. The spatially correlated background noise $\boldsymbol{b}(t_i)$ is simulated as:

$$\boldsymbol{b}(t_i) = {}^{c}\sqrt{\boldsymbol{R}}\,\boldsymbol{b}'(t_i) \qquad (3\text{-}48)$$

where

${}^{c}\sqrt{\ }$ = Cholesky square root

$\boldsymbol{b}'(t_i)$ = 64-dimensional vector of readily simulated discrete, independent white Gaussian noise with statistics:

$$\begin{aligned} E\{\boldsymbol{b}'(t_i)\} &= \boldsymbol{0} \\ E\{\boldsymbol{b}'(t_i)\boldsymbol{b}'^T(t_j)\} &= \boldsymbol{I}\delta_{ij} \end{aligned} \qquad (3\text{-}49)$$

3.3.2 Low-Energy Laser Speckle Reflection Model. The low-energy speckle reflection model evolved through the work accomplished by Eden and Evans [8,9]. The model makes no attempt to simulate the detailed physical phenomena associated with the speckle return of the reflected laser from the plume or hardbody. Rather, the model simulates the reflectivity information from the hardbody speckle return which would be derived by speckle detection circuitry.

The low-energy laser speckle reflection model simulates a linear measurement to the Kalman filter for estimating the offset distance from the plume intensity centroid to the hardbody center-of-mass along the vehicle's FLIR image plane velocity vector. The first attempt to model the laser speckle return consisted of the hardbody represented as a rectangle with a binary-valued reflectivity function, which provided a binary indication of the hardbody whenever successful interception by the laser beam occurred [8]. With this model, speckle reflection information was equally obtained over the entire vehicle. This was followed by an enhanced, 3-dimensional, reflectivity model which accounted for the realistic distribution of the laser speckle return according to the curvature and aspect angle of the hardbody [9]. The 3-dimensional model is employed for this research since the Doppler return is also a function of reflectivity [14,43,44,50]. The following subsections discuss the development of Evans' 3-dimensional hardbody reflectivity model and introduce the plume reflectance model.

3.3.2.1 The Hardbody Reflectivity Model. The 3-dimensional reflectivity model was developed by Evans [9] based upon his analysis of empirical data obtained from the 6585th Test Group, Holloman AFB, New Mexico [10]. The data illustrates the return power (expressed in decibels-square meters) as a function of radar cross section (RCS) from a 20 x 249 inch cylinder with hemispherical endcaps as it was rotated longitudinally in the plane of the radar source. (RCS is defined as the projected area of a metal sphere which would return the same echo signal as the target, had the sphere been substituted for the target [43].) The data showed peak values at 90° and 270°, where the cylinder was orthogonal to the line of sight, and sharp dropoffs in reflection as the angle deviated from the orthogonal condition. The reflectivity model, shown relative to the FLIR image plane in Figure 3.4, modifies the previous rectangular model to include 29 discrete-weighted line segments along the length of the model. Two functions define the hardbody reflectivity model: the cross-sectional function and the longitudinal function.

Each discrete-weighted line represents a cross-sectional reflectivity function which duplicates the empirical data from Holloman. The reflectivity function models the curvature by defining the strength of the reflected signal at each discrete line, where the amplitude of the reflected signal is highest along the missile centerline and discretely tapers towards the hardbody sides in 0.1 meter increments. The discrete implementation of the cross-sectional reflectivity function for the simulation is shown in Figure 3.5. Note the peak reflection of the cross-sectional reflectivity function's center is represented by an arbitrary value of 50 units of reflection magnitude [9]. The remaining line segments are scaled accordingly to match the empirical data. The reflectivity function also yields zero reflection for those portions of the original rectangle far from the missile centerline, so the effective reflective area of the hardbody is less than that of the binary model.

The angle γ, defined as the angle between the inertial velocity vector and the FLIR plane, is utilized by the longitudinal reflectivity function to provide a scaling factor of the total reflection function if the missile centerline is oriented other than normal to the FLIR plane. Similar to the

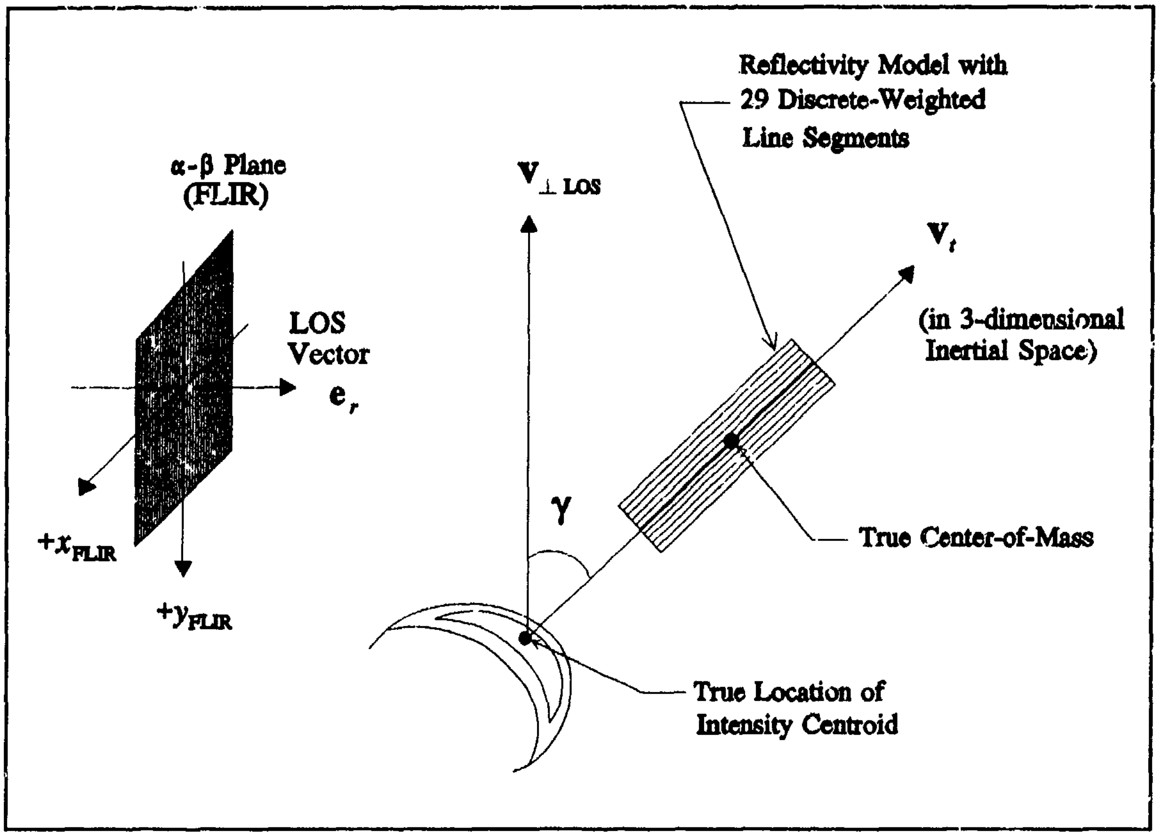


Figure 3.4 3-d Hardbody Reflectivity Model Relative to FLIR Image Plane

cross-sectional reflectivity function, the longitudinal function assigns a scaling factor to the reflected signal based upon the angular aspect of the target velocity.

Another factor in determining the received speckle reflection is the sensitivity level of the low-energy laser sensor. This sensitivity is represented in the simulation as a threshold limit below which the low-energy laser sensor cannot detect the reflection return. To illustrate the function of the sensor sensitivity factor, consider the hardbody at an aspect angle γ relative to the FLIR image plane. In this orientation, the maximum amount of reflection is obtained in the simulation by multiplying the peak reflection value (50 units of magnitude) by an appropriate scaling factor [9]. The sensitivity threshold function $\mu(\cdot)$ is defined as a function of a threshold

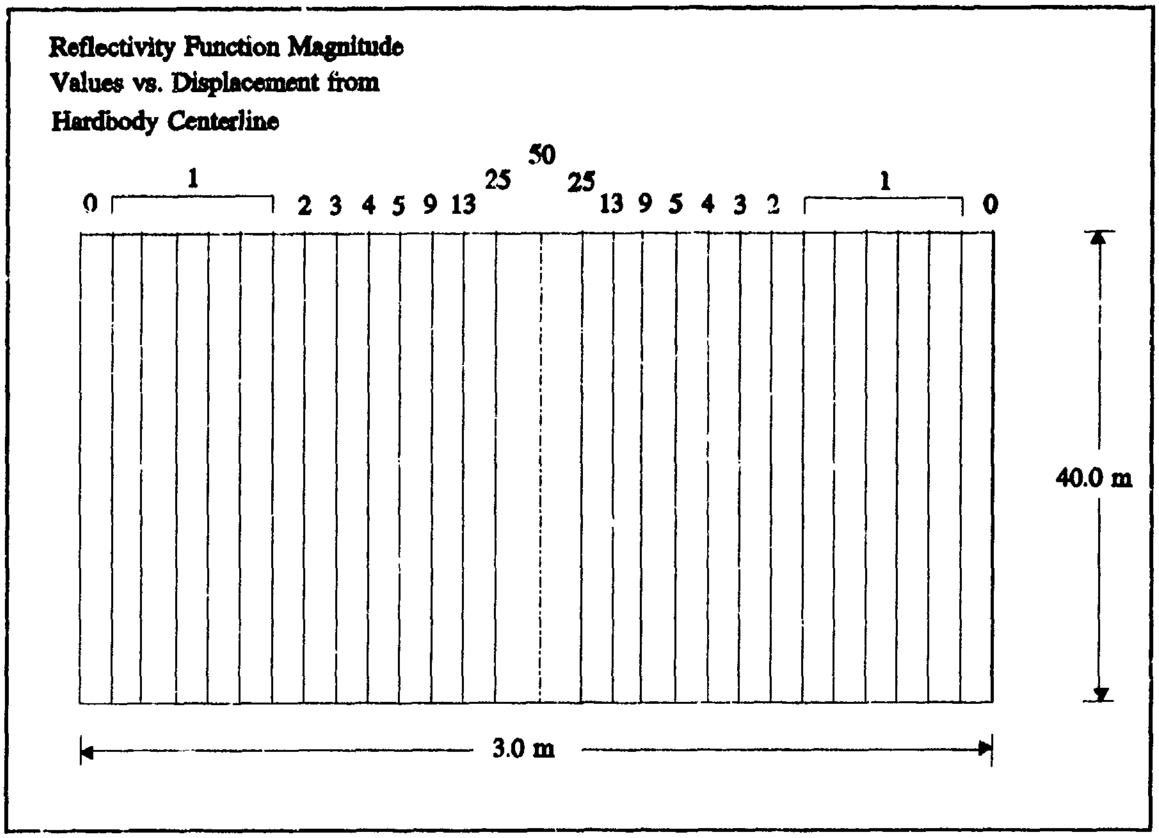


Figure 3.5 Discrete Implementation of Cross-Sectional Reflectivity Function [9]

reflection magnitude m_r. If a reflection $(\cdot)$ is less than m_r, the reflective output is clipped to zero (see defining equation for $\mu(\cdot)$ in the next paragraph). Therefore, $\mu(\cdot)$ represents the sensor's ability to discern a target's return signal [9].

The total reflectivity function is given by [9]:

$$R_T = \sum_{i=1}^{n} \mu\,[A_i F(\gamma)] \tag{3-50}$$

where

R_T = total reflectivity received by the low energy sensor

n = number of line segments crossed by laser scan

$\mu(\cdot)$ = sensitivity threshold function of low-energy sensor:

$$\mu(m_r) = \begin{Bmatrix} m_r \text{ if } m_r \geq \text{threshold} \\ 0 \text{ if } m_r < \text{threshold} \end{Bmatrix}$$

A_i = cross-sectional reflectivity function reflection amplitude of the i^{th} discrete line segment

$F(\gamma)$ = longitudinal reflectivity function, where γ is the angle between target v and the α-β plane

As the hardbody traverses along its trajectory in 3-dimensional inertial space, the projection of its motion onto the 2-dimensional FLIR image plane generates the corresponding propagation of the first two states in the truth model. Similarly, to simulate the center-of-mass measurements in terms of FLIR plane variables, the hardbody model is also projected onto the 2-dimensional FLIR plane. Referring to Figure 3.6, the geometry for projection is described by:

$$ML_{FLIR} = ML_{Actual}\cos\gamma \tag{3-51}$$

where

ML_{FLIR} = FLIR plane projection of missile length

ML_{Actual} = true missile length in pixels

γ = angle between v_t (velocity vector of the target) and the FLIR plane

Similarly, since the hardbody longitudinal axis is assumed to be aligned with the velocity vector (along which the offset is aligned), the offset between the hardbody and the plume is scaled by the same factor when projected onto the FLIR plane. Since the missile is cylindrical, the projection of the missile diameter onto the FLIR plane is equal to its diameter. Once the projection is accomplished, the hardbody is located on the FLIR plane by offsetting the hardbody's center (midway between the projected endpoints) from the truth model intensity centroid along the truth model velocity vector by $[(Offset\ distance_{actual})\cos\gamma]$.

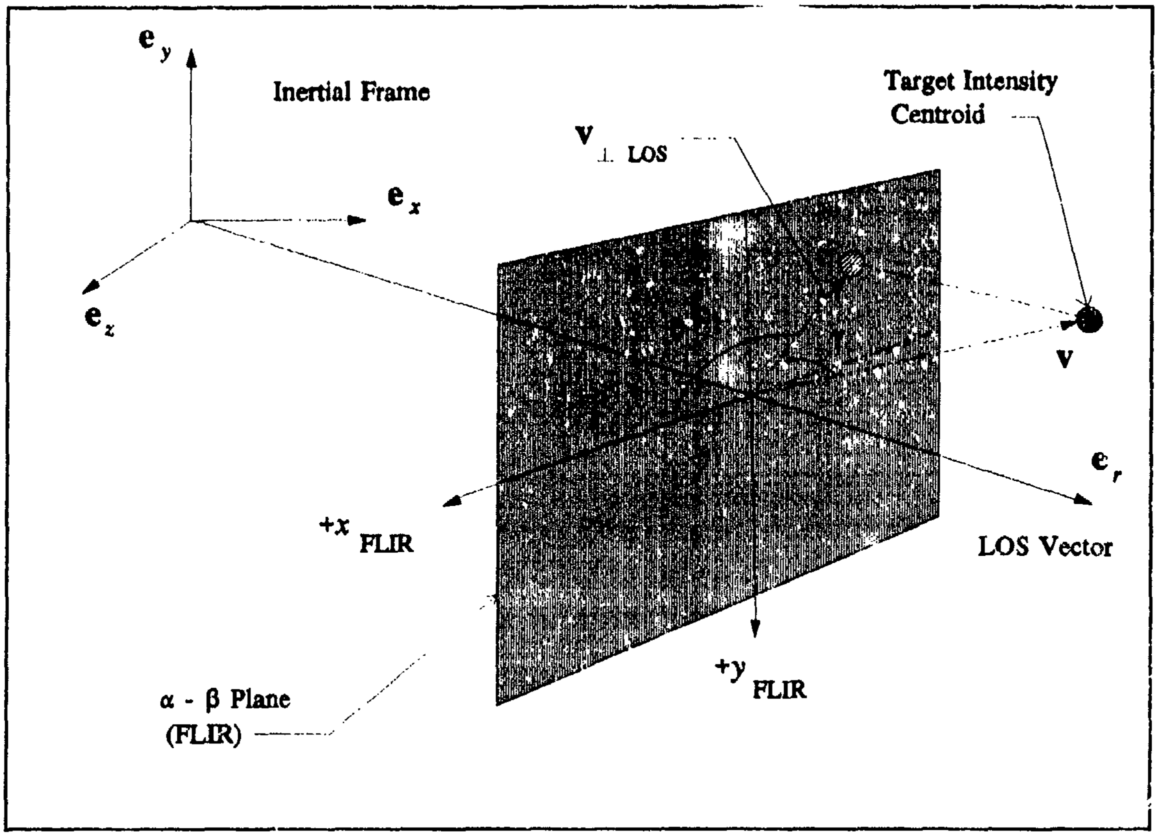


Figure 3.6 Projection Geometry onto FLIR Image Plane

The subtended arc of the low-power laser beam is simulated as a rectangle with the smaller side represented as the finite width of a dithered laser beam after it has traveled 2000 kilometers. Shown in Figure 3.7 are the ideal conditions for the laser scan. (Generally, the filter estimates of the intensity centroid position, the orientation angle, and the velocity vector are not equal to the truth model values.) One end of the long centerline of laser scan rectangle is located at the estimated intensity centroid, positioned at the center of the FLIR FOV. The other end of the laser scan rectangle is taken as three times the truth model offset distance between the intensity centroid and the hardbody center-of-mass (3 x 87.5 = 262.5 meters or 8.75 pixels) to ensure the laser scan is long enough to intercept the hardbody, despite the effects of "pogo". The

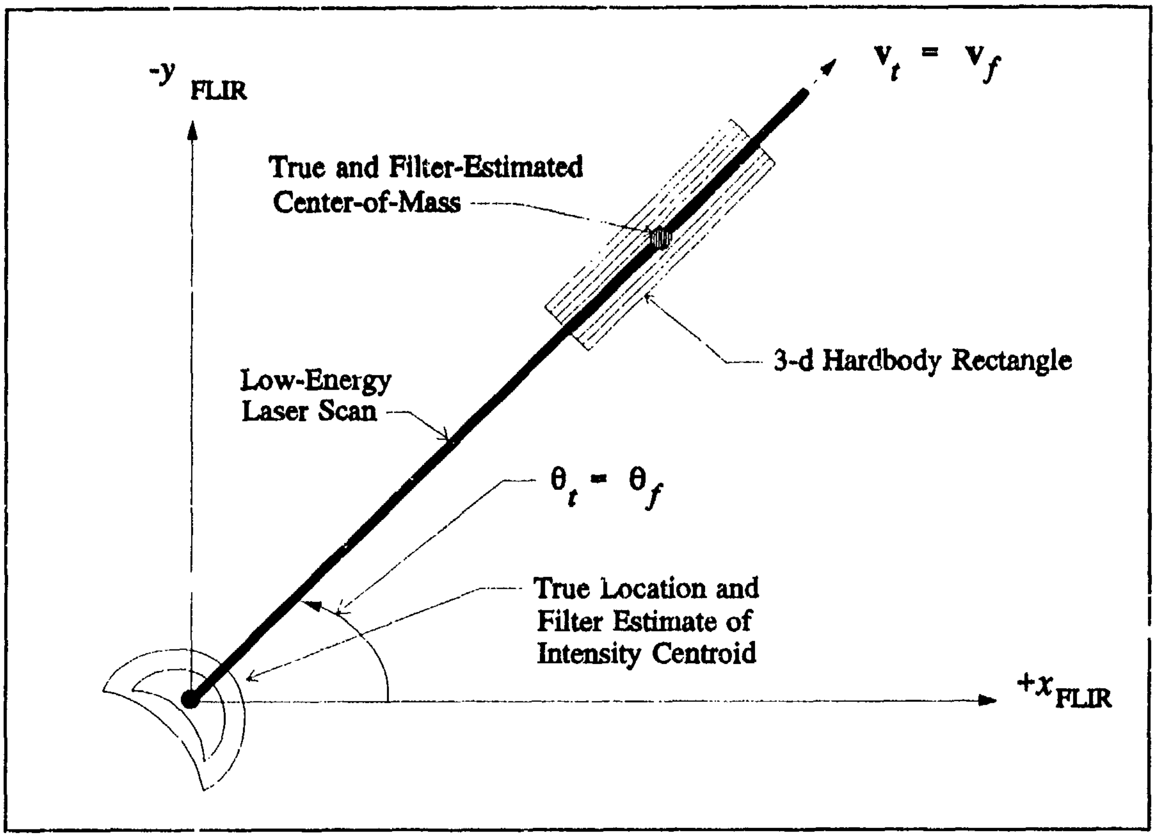


Figure 3.7 Ideal Low-Energy Laser Scan

second endpoint of the laser rectangle along its centerline is given as:

$$\begin{aligned} x_p &= x_c + L\cos\theta_f \\ y_p &= y_c - L\sin\theta_f \end{aligned} \tag{3-52}$$

where

x_p, y_p = the FLIR plane coordinates of the second end of the centerline of the laser rectangle

x_c, y_c = the FLIR plane intensity centroid coordinates

L = length of the laser rectangle

θ_f = six-state (FLIR) filter estimate of velocity orientation angle

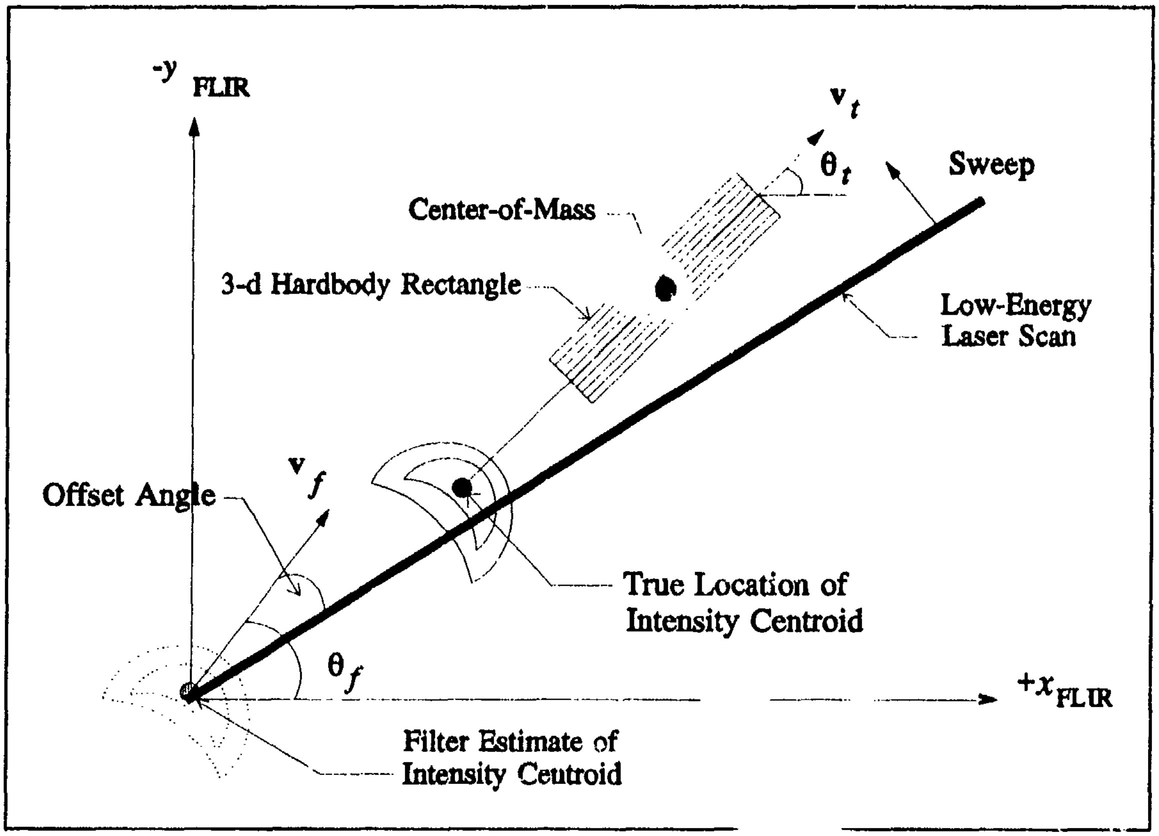


Figure 3.8 Sweep Technique of L [illegible] can

As mentioned earlier, the FLIR filter's imprecise centering of the inten [illegible] centroid caused inadequate hardbody illumination rates by the laser scan in the original research by Eden [8]. (The estimated velocity vector, and thus the estimated orientation angle, θ_f, were estimated precisely, however.) As a result, an *ad hoc* sweep routine was developed, shown in Figure 3.8, that offsets the initial laser scan clockwise from the estimated velocity vector. The laser scans are swept counterclockwise in order to assure illumination of the entire body. Evans found that, without pogo, a 30° offset was required, and 35° with pogo applied [9].

3.3.2.2 Plume Reflectance Model. Prior to Herrera's research, the concept of illuminating the missile hardbody with a low-energy laser and analyzing the speckle return (also called backscatter radiation) was predicated upon the assumption that the missile plume would not possess any backscatter properties or possess a speckle return similar to the hardbody's, when illuminated by a low-energy laser [14]. The laser scan travels along the intensity centroid's velocity vector until a speckle return is received signifying the start of the metallic hardbody. The scan continues along the hardbody until no backscatter exists, signaling the end of the hardbody, and thus information is provided to calculate the center-of-mass. However, experimental data confirmed the presence of plume reflectance from solid-propellant rocket motors [2,34] which significantly alters the previous conception.

Experimental programs at the Arnold Engineering and Development Center (AEDC), in Tennessee, have observed and measured laser backscatter radiation from the exhaust plume of a solid-propellant rocket motor [34]. The measurements of the plume's backscatter radiation were found to be on the same order of magnitude and comparable to that of a hardbody [2], due to aluminum particles and other substances in the plume. During the STARLAB flight experiment, which collected plume data under actual flight conditions, a rocket booster and its exhaust plume were "painted" by a low energy laser. Video recordings of the flight experiment showed the randomized appearance and low-frequency oscillation of the plume's reflectance [2,3]. The existence of plume reflectance creates an ambiguity that impedes the precision tracking necessary to define the plume/hardbody interface. The plume reflectance causes a bias in the estimated hardbody location, biased longitudinally toward the plume.

Since for this thesis, the Doppler measurement model was utilized instead of speckle, the offset measurements from the LEL were assumed to be unbiased (see the next section). However, this section has been included in the thesis description for continuity and, since the bias effect is still implemented in the software if the speckle return model is used, it is retained as reference for future researchers.

The purpose of the plume reflectance model is to simulate the presence of plume backscatter radiation and its effect upon the offset measurement. Figure 3.9 depicts the reflectance from both the plume and hardbody, as observed in the STARLAB flight experiment. From the viewpoint of the speckle return sensor, the plume reflectance has the effect of elongating the apparent missile hardbody in the direction of the plume. The plume reflectance model simulates the hardbody elongation by applying a bias to the offset measurement in the direction of the

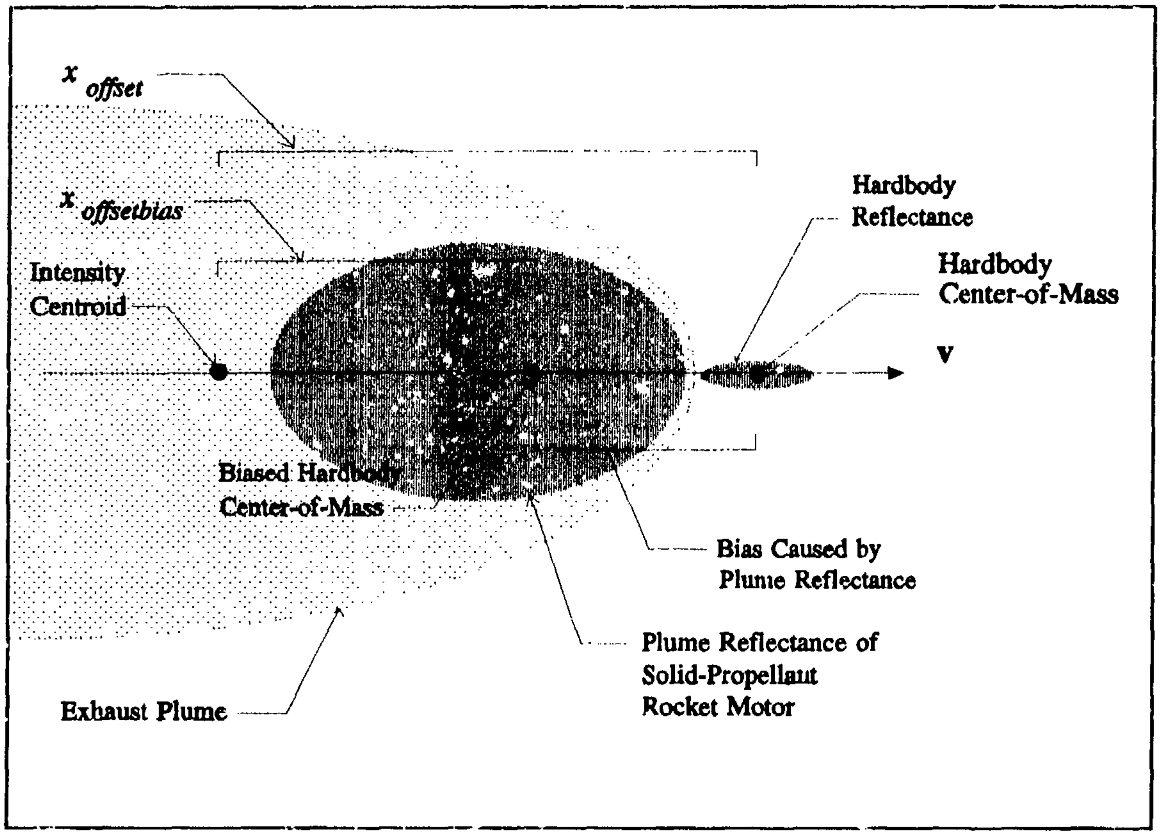

x_{offset}
$x_{offsetbias}$
Intensity Centroid
Hardbody Reflectance
Hardbody Center-of-Mass
V
Biased Hardbody Center-of-Mass
Bias Caused by Plume Reflectance
Plume Reflectance of Solid-Propellant Rocket Motor
Exhaust Plume

Figure 3.9 Biased Offset Measurement Caused by Plume Reflectance

elongation, defined as in the opposite direction of the estimated velocity vector. In the simulation, the model first receives the offset measurement as determined by the low-energy speckle reflection model. The biased measurement, $x_{offsetbias}$, is formed by converting the bias into pixels, projecting it onto the FLIR plane, and subtracting it from the original offset measurement. The biased offset measurement is then provided to the filter for its update. The plume reflectance model is given by:

$$x_{offsetbias} = x_{offset} - \left(\frac{b}{Rk_p} \right) \cos\gamma \tag{3-53}$$

where

$x_{offsetbias}$ = biased offset measurement due to plume speckle reflectance
x_{offset} = offset measurement from the low-energy reflectivity model, without plume speckle reflectance effect
b = bias value
R = range
k_p = pixel proportionality constant (15 μrads/pixel)
γ = angle between 3-dimensional inertial space velocity vector and the FLIR image plane

The randomized nature of the plume's reflectance is modeled as a percentage of time that appearance of the bias occurs. A random number generator, of uniform density output, provides the logic to turn the bias "on and off" according to the percentage selected. In correspondence with Phillips Laboratory personnel, a bias of approximately 25-30 meters with an appearance percentage of 90 - 95% was observed during the STARLAB flight experiment [3].

3.3.3 The Doppler Measurement Model. The Doppler measurement model simulates the offset measurements that are obtained by exploiting the differences between hardbody and plume-induced Doppler returns. As with the laser speckle return research of Eden and Evans, the modeling of the actual physical properties of the Doppler phenomenon will not be attempted. Instead, modeling efforts will entail simulating the information that would be available from Doppler detection circuits as measurement data for the Kalman filter. The following subsections briefly introduce and describe the basic concepts of the Doppler phenomenon, as applicable to the properties of the hardbody and plume-induced Doppler returns. The treatment of the Doppler phenomenon is not intended to be rigorous and reflects the level of understanding necessary to appreciate the manner with which the Doppler returns are employed to generate an offset measurement relative to the intensity centroid. For a rigorous development of the Doppler phenomenon, the interested reader is referred to *Principles and Practice of Laser-Doppler Anemometry* by Durst, F., A. Melling, and J. H. Whitelaw [7], and *The Doppler Effect* by Gill, T. P. [12].

3.3.3.1 The Doppler Effect. Many define the Doppler effect as a shift in the frequency of a wave radiated, reflected, or received by an object in motion [43,44]. From a radar, Doppler shifts are produced by the relative motion between the radar and the target. The radar may be a pulsed, coherent laser beam that propagates the electromagnetic energy to "paint" the target of interest. If the target is in motion and illuminated by a low-energy laser, the returned signal (or backscatter) is represented as a time-delayed, Doppler-shifted version of the transmitted signal, wherein the amount Doppler shift is proportional to the reflecting target's range rate relative to the laser transmitter [43,44]. A continuous transmitted signal is given as:

$$E_t = E_o \cos(2\pi f_o t) \tag{3-54}$$

For this transmitted signal, the echo signal from a moving target will be [43]:

$$E_r = kE_o \cos[2\pi(f_o \pm f_d)t + \phi] \tag{3-55}$$

where

E_o = amplitude of transmitted signal

E_r = reflected signal

k = an attenuation constant that represents losses incurred during propagation

f_d = Doppler frequency shift

f_o = transmitted frequency

ϕ = a phase shift, dependent upon the range of detection

Figure 3.10 shows the frequency spectrum of the echo signal, shifted from the transmitted frequency, f_o, by the Doppler shift, f_d, given by [43]:

$$\pm f_d = \pm\frac{2v_r}{\lambda} = \pm\frac{2v_r f_o}{c} \tag{3-56}$$

where

v_r = relative velocity of target with respect to transmitter

λ = transmitted wavelength

c = velocity of propagation (3×10^8 m/s)

The relative velocity, v_r, is expressed as:

$$v_r = v \sin\gamma \tag{3-57}$$

where

v = target velocity in 3-dimensional inertial space

γ = angle between the target trajectory and plane perpendicular to the laser LOS (i.e., FLIR plane); see Figure 3.6

The plus sign associated with the Doppler frequency shift applies if the distance between target and transmitter is decreasing (approaching target), and conversely, the minus sign applies if the distance is increasing (receding target).

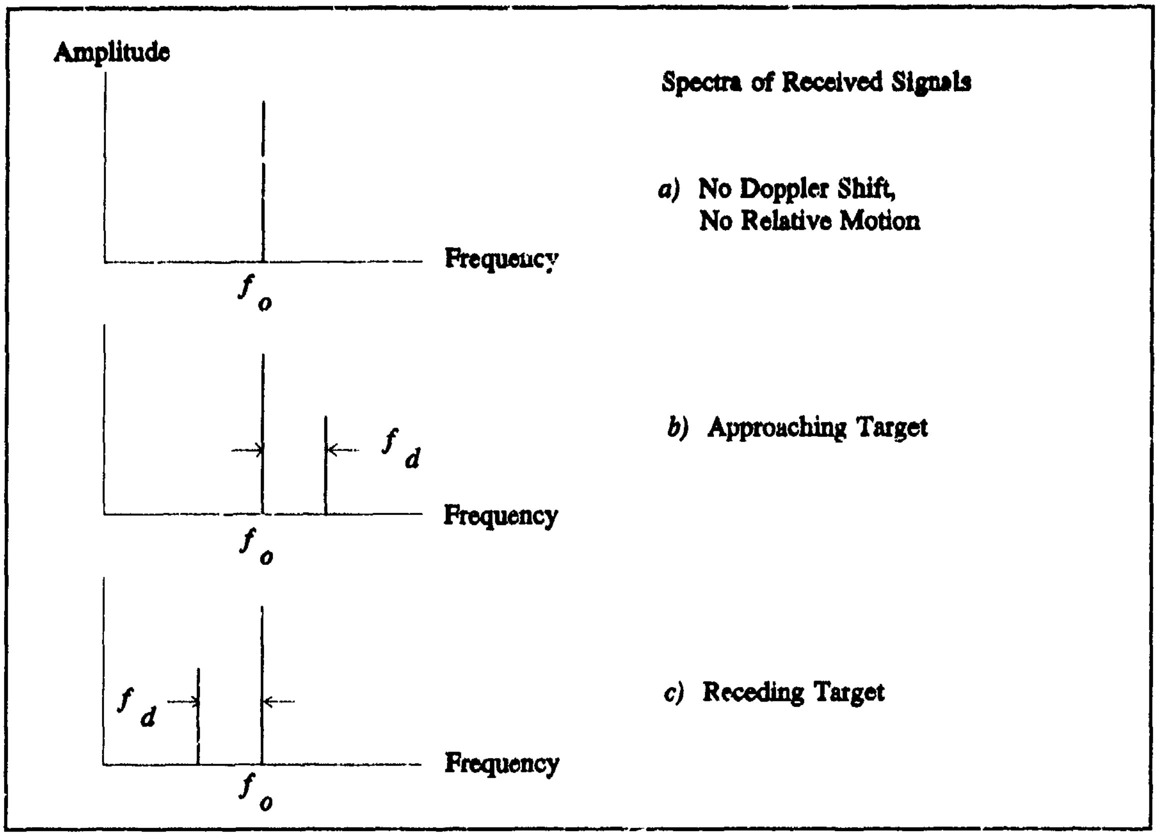


Figure 3.10 Spectra of Received Signals [43]

As shown in Figure 3.10, the frequency spectrum of a continuous reflected sinusoidal signal appears as a straight vertical line. The scenario proposed by the Phillips Laboratory calls for a pulsed and coherent laser beam to illuminate a ballistic boosting target [3]. Both these laser properties have an impact upon the nature of the returned spectrum.

For illustration purposes, Figure 3.11 shows a train of independent pulses having a pulse width (PW) of 0.001 seconds and a constant pulse repetition frequency (PRF), along with its associated frequency spectrum. Because the pulses are "on" a fraction of the time, the amplitude of the frequency spectrum decreases but is still centered at f_o. The total power is in fact distributed over a band of frequencies extending from 1000 Hz below f_o to 1000 Hz above it, for a null-to-null bandwidth of 2 KHz. The bandwidth (i.e. spectrum spread), is inversely proportional to the pulse width and is given by [44]:

$$BW_{nn} = \frac{2}{\tau} \tag{3-58}$$

where

BW_{nn} = null-to-null bandwidth
τ = pulse width (sec)

By coherence is meant a consistency, or continuity, in the phase of a signal from one pulse to the next [44]. The term ϕ in Equation (3-55) represents the phase shift which is a function of the range during detection. Figure 3.12 illustrates the difference between the frequency spectrum of a coherent signal and a non-coherent signal. With non-coherent transmission, the signal's central spectral lobe is spread over a band of frequencies. In contrast, the spectrum associated with coherent transmission shows the signal appearing at many points. Its spectrum, in fact, consists of a series of evenly spaced lines, wherein the interval between the spectral lines equals (1/PRF) [44]. Further comparison reveals that the coherent frequency spectrum is stronger (having a higher amplitude) than the non-coherent signal because the energy has been concentrated into a few narrow lines. In addition, the envelope within which these lines fit has the same shape ([sinx]/x]) and the same null-to-null BW (2/τ) as the spectrum of the non-coherent signal.

3.3.3.2 Hardbody Doppler Return. At a range of 2000 kilometers, the missile hardbody can be defined as a smooth, dense single point target. Any rotational motion of the hardbody about its longitudinal axis is assumed much less than the hardbody's velocity, and is considered negligible. It is further assumed that the target hardbody's velocity remains constant over the

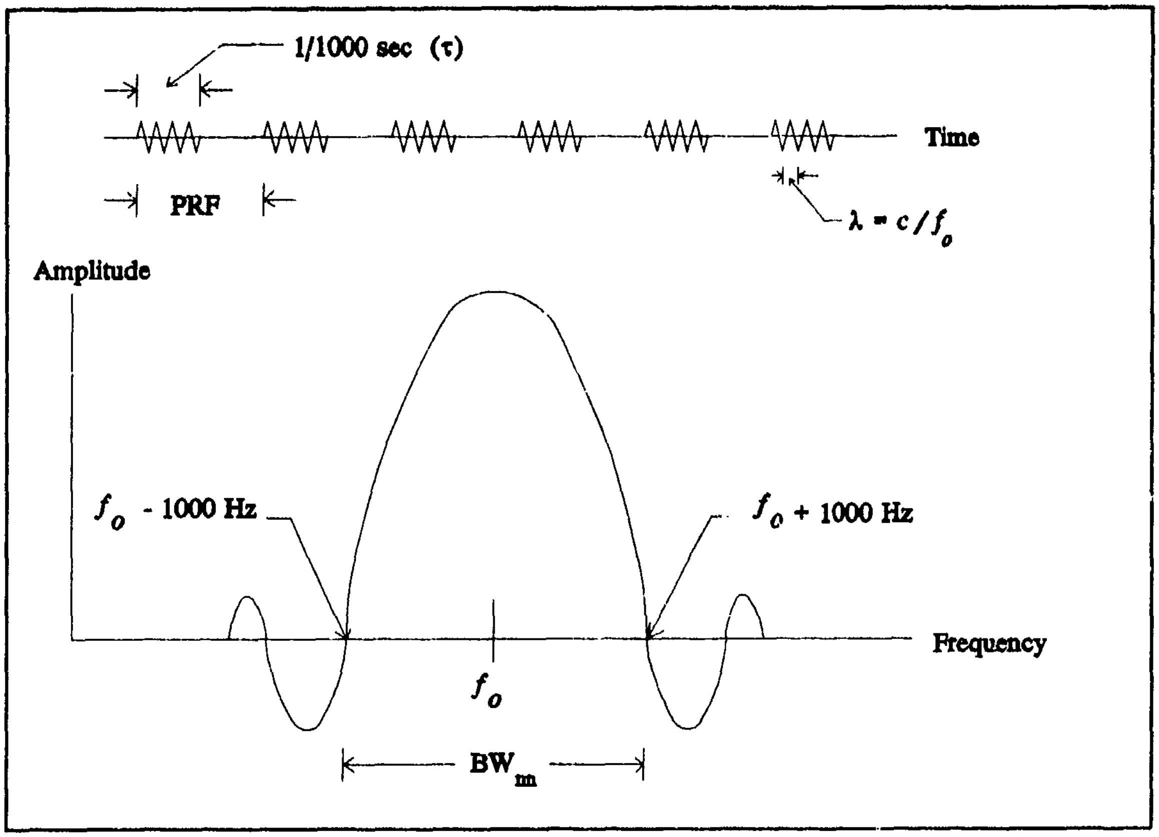


Figure 3.11 Pulsed Signal Frequency Spectrum [44]

duration of a transmitted pulse. With such a target, the spectrum of the return will have a bandwidth that closely approximates (2/τ), and centered about the Doppler-shifted frequency corresponding to the range rate.

3.3.3.3 Plume Doppler Return. The case of the exhaust plume can be represented as the situation in which numerous point targets are imaged. The plume can be described as a randomly distributed array of point targets which are dispersed in range and velocity. The plume particulates are small (submicron in size), nonspherical and nonhomogeneous, and their size and spatial distribution vary strongly with the radial distance from the plume axis [14,51]. Typically, larger particles are concentrated near the plume's symmetry axis, and in contrast to the hardbody,

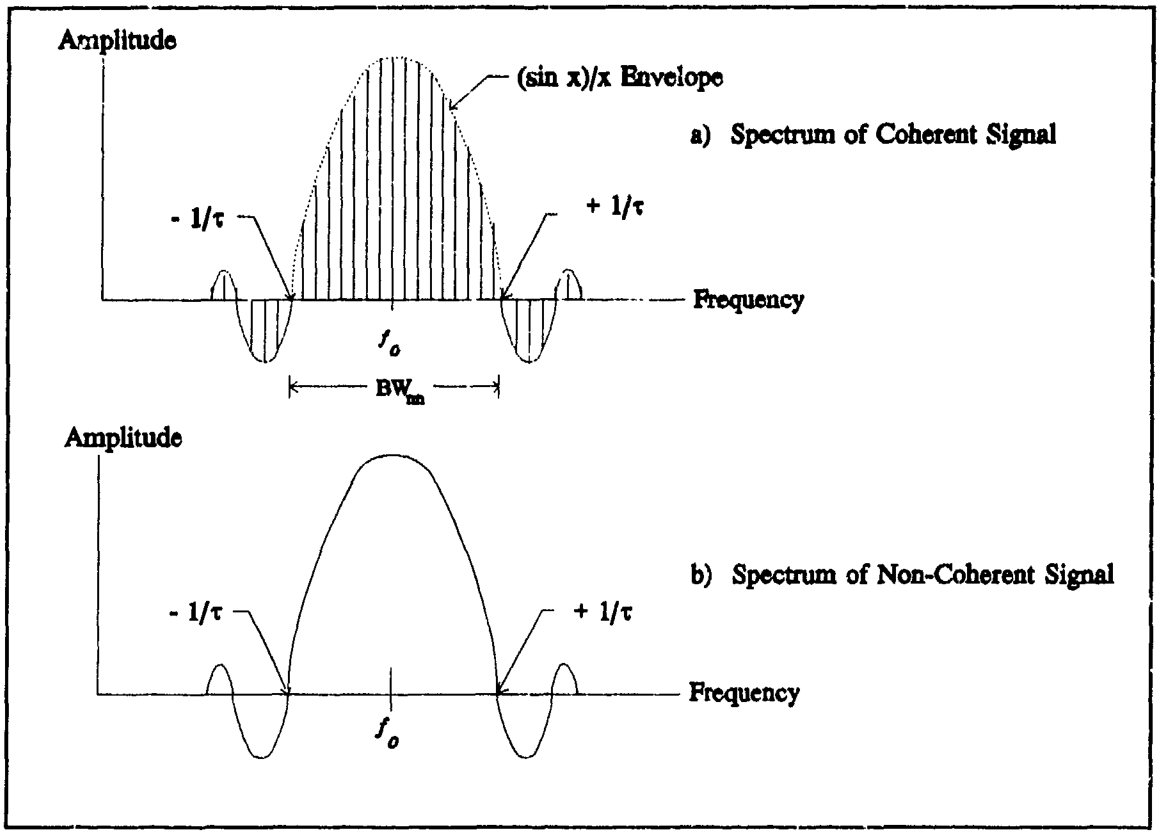


Figure 3.12 Spectra of Coherent and Non-Coherent Pulsed Signals [44]

the numerous exhaust plume particles exhibit numerous velocity orientations over the duration of a laser pulse.

When the laser beam illuminates an infinite number of point targets, the superposition of each particle's backscatter radiation within the laser beamwidth will form the resultant return [50,51]. Thus, the Doppler frequency spectrum will be quite broad, due to the numerous Doppler shifts of the numerous plume particulate velocities [3,14]. This Doppler spreading of spectral lines arises from the fact that backscatter from a particulate will be shifted in frequency in a manner depending on the approach or recession of the particulate. The plume experimental programs at AEDC have observed and measured plume Doppler reflectance frequency spectrums

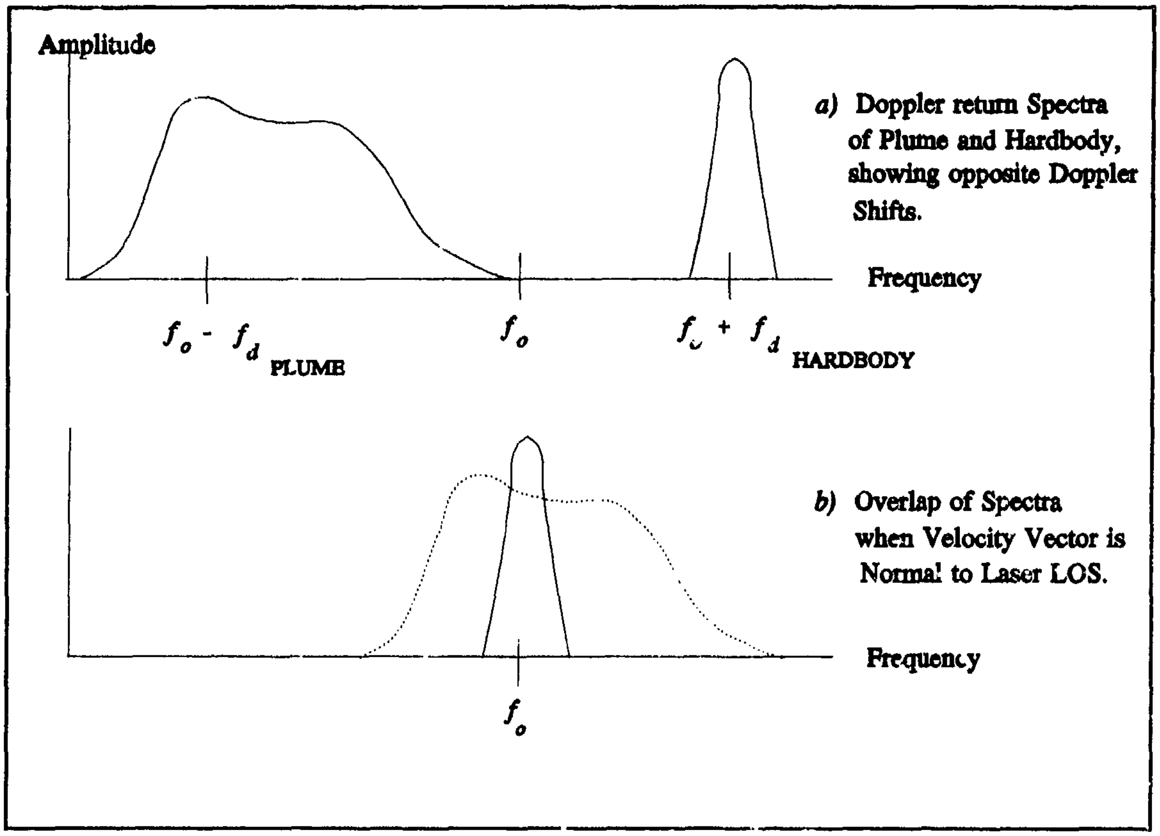


Figure 3.13 Spectra of Plume and Hardbody-Induced Doppler Returns

with null-to-null BWs of 2 - 5 GHz [32]. This sharply contrasts the hardbody-induced return whose spectrum null-to-null BW equals $2/\tau$, with an order of magnitude in MHz. However, one other significant difference exists between the hardbody and plume-induced Doppler returns.

Generally, the velocity of the plume will be oriented 180 degrees from the hardbody's velocity [3,14]. This is shown in Figure 3.13(a), where the respective Doppler frequency shifts will always be opposite in sign. A majority of the observed plume particles would have a relative radial velocity towards the tracker and the resultant return would have a negative Doppler frequency shift. Conversely, the hardbody as shown is receding from the tracker and will thus exhibit a positive Doppler frequency shift. Hence, by exploiting the two differences in plume and hardbody-induced Doppler returns, precise tracking and definition of the plume/hardbody interface can be realized.

However, the angle γ, of which the relative velocity v_r is a function, has an impact upon the discernability between the plume and the hardbody-induced Doppler shifts. Referring to Equation (3-57), as γ approaches 0°, where the plume and hardbody velocity vectors become orthogonal to the LOS vector, the radial velocity relative to the tracker approaches nil and no Doppler shift is produced. Figure 3.13(b) shows that, under these circumstances, the return spectra of the plume and hardbody converge towards the transmitted frequency and eventually overlap, obscuring most of the hardbody-induced Doppler return. This imperfect ability to detect the hardbody spectrum, as distinct from the plume spectrum, will be addressed in the next section which develops the Doppler measurement model.

The measurement modeling approach taken by this thesis is to consider the Doppler return of the hardbody significantly distinctive from that of the plume. The Doppler detector must be designed to filter out the broader plume return and only pass the hardbody return, a function achievable with a Doppler matched filter design [14,43]. This vital concept signifies that the Doppler truth measurement model can neglect the plume's Doppler return and solely simulate the hardbody-induced Doppler return. Although there may be instances of no apparent distinction between the plume and hardbody spectra, these occurrences will be embodied in a probability-of-miss parameter (P_m), to be discussed later.

Since Doppler information is obtainable from backscatter radiation, which includes the speckle return [43,44], the 3-d hardbody reflectivity model is utilized in this modeling approach. However, in contrast to the laser speckle return measurement model, the biasing effect caused by the plume's reflectance is no longer applicable and is not incorporated into the Doppler measurement model. As a result, the center-of-mass measurement and offset measurement generated by the Doppler measurement model will simulate the true offset measurement, x_{offset}, for the filter.

3.3.3.4 Doppler Measurement Noises. According to Herrera, in a study sponsored by the Phillips Laboratory, Dr. Paul McManamon investigated feasible and implementable wavelengths to illuminate the plume and hardbody, while meeting the space tracking scenario requirements [14]. His choice of wavelengths, based upon ranges, power requirements, hardbody temperatures, and tracking accuracies, range from 0.53 to 15 μm. For this study, the shortest wavelength 0.53 μm was selected for use in sensitivity analysis. The tracking inaccuracies associated with this wavelength is adopted in the Doppler measurement model to corrupt the offset measurements realistically.

The tracking accuracy for a laser beam is a function of the amount of power, or amplitude, of the return signal. The return signal, in turn, is dependent upon several variables, among which are the target's radar cross section (RCS) and the location of the target in the laser beam [14,43,44]. A target ideally located in the center of the laser beam reflects the maximum return signal (i.e., optimum signal-to-noise ratio, SNR). If the target falls off to the side of the beam's center, then less energy hits the target. The degree of tracking accuracy then becomes a question of, how far off to the side can a target be to reflect the signal at an acceptable level?

Dr. McManamon addressed this issue [14] by first defining the acceptable beam diffraction limit as the angle within the 3 db power points of the laser beam. He defines the diffraction limit as:

$$\theta_{3db} = 1.08\frac{\lambda}{d} \tag{3-59}$$

where

θ_{3db} = half angle defined from beam center to half-power points, in radians
λ = wavelength, in meters
d = radar aperture, in meters

One then determines the acceptable level of signal loss within the θ_{3db} limits. In Dr. McManamon's assessment, a 10% loss can be tolerated, and he determined that this loss is reflected by decreasing the diffraction limit by a factor of 2.667 [14]. Equation (3-59) yields:

$$\theta_B = \frac{\theta_{3db}}{2.667} \tag{3-60}$$

where

θ_B = allowed diffraction limit for 10% signal loss

The measurement noise for the Doppler measurement model thus consists of the tracking angle errors, in pixels, as a function of the diffraction limited beam and acceptable signal-to-noise ratio (SNR). Inasmuch as SNR is a design parameter, Herrera's study [14] included the following values of SNR for the sensitivity analysis: 10, 8, 6, and 4. The relationship is given as [14]:

$$\theta_T = \frac{\theta_B}{3\sqrt{\mathrm{SNR}}}\frac{1}{k_p} \tag{3-61}$$

where

θ_T = rms tracking angle errors in pixels
θ_B = beam diffraction limit

SNR = signal-to-noise ratio

k_p = pixel proportionality constant, 15 μrads/pixel

In addition to providing the offset measurement, the Doppler measurement model also simulates a return signal probability-of-miss, P_m. The probability-of-miss encompasses two cases. First, the probability-of-miss takes into account the situation in which the hardbody is illuminated by the low-energy laser, but the return is not detected due to attenuation of the returning signal as it propagates the 2000 kilometer range, beam-bending as a result of atmospheric distortions (the intended location of the laser scan should have illuminated the target, but bending of the beam resulted in no intersection with the target); or due to signal losses (i.e., high sensor sensitivity threshold; refer to Section 3.3.2.1) within the receiving equipment. In this case, a loss of speckle information would also result. Secondly, in Equation (3-51), it was shown that the relative velocity is a function of γ, such that no Doppler shift occurs if the target's velocity is normal to the transmitter's LOS. Hence, as shown in Figure 3.13, as γ approaches 0°, both the broadened plume-induced Doppler spectrum and hardbody-induced spectrum will converge and overlap. The two spectra will become more indistinguishable, perhaps rendering detection of the hardbody's Doppler return impossible.

The simulation of the probability-of-miss is similar to the technique employed by the plume reflectance model. A random number generator, with a uniformly distributed output, also provides the logic to turn the hardbody laser backscatter "on and off." Figure 3.14 shows the detection characteristic for a known signal. The graph present a set of parametric curves that give the probability-of-detection, P_d, as functions of peak signal-to-noise ratio (SNR) for various values of probability-of-false alarm, P_{fa}. P_{fa} is defined as falsely indicating the presence of a return signal when none exists [43]. Both P_d and P_{fa} are specified by the system requirements; the radar designer computes the probability-of-false alarm and, from Figure 3.14, determines the minimum detectable signal. A range of 70 - 99 percent probability-of-detection is representative of current Doppler detection equipment capabilities with the tracking scenario [14].

3.4 Truth Model Parameters

The discussions in the previous sections introduced some of the truth model parameters used in the simulation. The purpose of this section is to provide a consolidated listing of the parameters and initial conditions of the truth model.

3.4.1 Target Trajectory Initial Conditions. The initial conditions of the target inertial position, velocity, and velocity vector orientation angle, θ, are as follows:

$$\begin{aligned}
e_x &= 27{,}000 \text{ meters} \\
e_y &= 100{,}000 \text{ meters} \\
e_z &= 2{,}000{,}000 \text{ meters} \\
v_x &= -2500 \text{ meters/sec} \\
v_y &= 4330 \text{ meters/sec} \\
v_z &= 0 \text{ meters/sec} \\
\theta &= 60^\circ
\end{aligned}$$

3.4.2 Target Model, Dimensions, and Orientation. The target plume consists of a crescent-shaped intensity function formed from the difference of two bivariate Gaussian intensity functions. Each Gaussian function is modeled with elliptical constant-intensity loci with an aspect ratio of 1.5, and a semi-minor axis of one. For this thesis, Evans' 3-dimensional reflectivity model is used to model the hardbody. The hardbody length is 40 meters (1.33 pixels) and 3 meters (0.1 pixels) wide. The offset distance of the hardbody center-of-mass from the intensity centroid is 87.5 meters (2.92 pixels), a carryover from the previous thesis. For the simulation, the intensity centroid and the hardbody longitudinal axis are aligned with the velocity vector, and the hardbody has zero sideslip and zero angle-of-attack.

3.4.3 Intensity Functions. The two Gaussian bivariate intensity functions, shown in Figure 3.3, are centered at 65 and 110 meters behind the missile. Each intensity function has a maximum intensity value of 20 intensity units.

3.4.4 Atmospheric Jitter. The variance and mean squared value for the atmospheric jitter in both FLIR directions is 0.2 pixels^2.

3.4.5 Bending/Vibration. From Equation (3-29), the values for the second-order bending/vibration model are as follows:

$$
\begin{aligned}
K_b^2 &= 5 \times 10^{-13} \text{ rad}^4/\text{sec}^4 \\
\zeta_b &= 0.15 \\
\omega_{nb} &= \pi \text{ rad/sec}
\end{aligned}
$$

3.4.6 Plume Pogo Characteristics. The size of the plume is on the order of 30 times the diameter of the missile at the altitudes of interest. The values below represent values of pogo oscillation as determined in previous research [40].

pogo oscillation = 0.1 - 10 Hz (nominal is 1 Hz)
pogo rms = 0.0112 - 1.120 pixels (nominal is 0.112 pixels)

For this research, the elemental filter in the AFIT software was tested against nominal conditions.

3.4.7 Spatially Correlated Background Noise. The rms value of $\boldsymbol{v}_{jk}$, the summed effect of the spatially correlated background noise $\boldsymbol{b}_{jk}$ and the FLIR sensor noise $\boldsymbol{n}_{jk}$, of Equation (3-45), equals one. This produces a SNR of 20.

3.4.8 Low-Energy Laser Speckle Return Measurement Dimensions. The low-energy scan is represented as a rectangle at the hardbody target. The scan length is 262.5 meters (8.75 pixels), which is three times the true model center-of-mass offset distance, and the scan width is 0.1 meters. The measurement noise associated with the speckle return was obtained by taking 1% of the hardbody's length, and converting to pixels, giving a variance of 0.000178 pixels^2 [9].

3.4.9 Plume Reflectance Model. From correspondence with Phillips Laboratory personnel [3], the bias utilized by the plume reflectance model is approximately 25 - 30 meters and appears 90 - 95% while the plume is illuminated during the boost phase. For the simulation, nominal values for the bias and rate of appearance are set at 25 meters and 90%, respectively.

3.4.10 Low-Energy Doppler Return Measurement Dimensions. The Doppler measurement noise rms tracking errors are functions of wavelength, radar aperture, and SNR. The previous thesis studied filter performance dependent upon the wavelength values of .53µm, 1.06 µm, 2.01µm, 4µm, 6µm, 8µm, and 10.5µm, with SNR values of 10, 8, 6, and 4, and probability-of-miss P_m values of 0.0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.10, 0.20, and 0.30. Since the purpose of this thesis was to incorporate pogo into the filter, the filter was given the benefit of the best possible Doppler measurements throughout the simulation. Therefore the conditions were set for a wavelength of .53µm, a SNR of 10 and a P_m value of 0.01. The radar aperture d of Equation (3-59) of .5 meters was carried over from the previous thesis.

3.4.11 Hardbody Reflectivity Measurement Model. The function $\mu(\cdot)$, in Equation (3-50), represents the sensitivity threshold of the low-energy laser return sensor. The threshold must be less than the magnitude of reflection, m_r (scaled according to the aspect angle γ), to detect the return from the hardbody. In the simulation, the value of the threshold is set to 0.0. This was to allow reception of measurements if any backscatter was received at all (perfect LEL receiving equipment).

3.5 Summary

This chapter presented the mathematical equations used to model the true environment. The truth model contains 14 states describing translational motion of the centroid in the FLIR plane due to target dynamics, atmospheric disturbances, bending/vibration of the platform and optics, and the pogo effect of the plume. The truth system is implemented in the AFIT software using the discrete-time dynamics/discrete-time measurement models presented in Section 3.2, and the continuous dynamics equations are utilized in the MSOFE simulations.

Section 3.3 delineated how the measurements are generated for the tracking scenario. The raw FLIR measurements are simulated by creating a composite plume intensity function on the 64-pixel array (the FOV) and then corrupting it with noise. The array data is then processed through the enhanced correlation algorithm to produce translational offsets of the centroid for the linear Kalman filter update. This section also outlined how the 3-dimensional missile hardbody is projected onto the FLIR plane and how the reflectivity function is determined based on scenario geometry. Finally, the use of the laser backscatter to formulate the Doppler measurement is also depicted. The same reflectivity model is used, but measurements are corrupted with rms angle tracking errors associated with a specific wavelength, radar aperture, and SNR, and there is no bias as there was in the speckle return measurements.

IV. Filter Models

4.1 Introduction

This chapter presents the filter dynamics and measurement models used for this thesis. All the structures described are reduced-order, simplified versions of the truth model. The dynamics model is separated into individual dynamics models for target dynamics, atmospheric disturbances affecting the plume intensity centroid location on the FLIR plane, and pogo affecting oscillation of the intensity centroid relative to the hardbody center-of-mass. Since the plant matrix (and corresponding discrete-time state transition matrix) and other matrices that form the models are of a block-diagonal structure, the various filters used in the MSOFE and AFIT software simulations can be assembled by including the portions associated with the desired states. Chapters V and VI describe the states that each individual filter configuration includes. This chapter also includes a description of the enhanced correlation algorithm which delivers "pseudo-measurements" of image intensity centroid offsets in the FLIR plane to the linear Kalman filter.

Three items of import should be remembered while reading the chapter. First, the target dynamics model for the filters differs from the truth model in that the deterministic driving input is zero. The filter models have four target dynamics states, with the velocity states being modeled as constants, and thus the derivatives of these states are set to zero (plus pseudonoise for filter tuning purposes). In contrast, there are only two position states in the truth model, driven by known (simulated) constant velocity values. The second noteworthy item is that the MSOFE filter models only incorporate FLIR measurements, and therefore do not deliver hardbody center-of-mass coordinates. Recall that the MSOFE simulations were only performed to gain insight into observability difficulties, so none of those simplified models incorporate the centroid/center-of-mass offset measurement. The third important item to remember is that the target dynamics states of the AFIT filter model describe the hardbody center-of-mass, whereas target dynamics states of the truth model and MSOFE models (and previous AFIT FLIR filter models [8,9,14,31,40,41]) describe the plume intensity centroid. This difference from previous research is because there is no separation of centroid location estimation and offset estimation, and all states have been combined into a single filter. Therefore, there is no "FLIR filter" and "center-of-mass offset filter" as in the previous three theses [8,9,14].

The AFIT software simulation filter model contains nine states (all four target dynamics states, a two-state approximated atmospheric jitter model, two pogo states, and the offset state). In the simulation, the filter performs an update based on the FLIR measurements. The updated position of the estimated location of the centroid is then used as the initial point of the laser scanning routine. The laser is swept in relation to the updated velocity vector to obtain an offset measurement via the Doppler return (as described in Section 3.3). The filter is then updated from the LEL measurement and the state vector is estimated for the next sample period.

4.2 Dynamics Models

The elemental AFIT filter employed in this study is a single nine-state filter combining the models that have been developed by AFIT students over thirteen years of research [8,13,14,31,33,40,41]. The filter consists of two hardbody center-of-mass position states, two hardbody velocity states, two atmospheric jitter position states (affecting centroid location in the FLIR plane), two pogo oscillation states (affecting centroid location relative to hardbody), and a centroid/center-of-mass offset state. The observability study filters used varying combinations of the following states:

- 4 or 6 target dynamics states (position, velocity, acceleration of *centroid* in x and y directions)
- 2 or 6 atmospheric jitter centroid position states
- 2 pogo states (offset from centroid equilibrium point and velocity)

The state vector for the desired filter can be assembled by including the required states from the following vector:

$$x = \left[\, x_t \; y_t \; v_x \; v_y \; a_x \; a_y \vdots x_a \; y_a \vdots x_p \; v_p \vdots x_o \,\right]^T \tag{4-1}$$

where

x_t = x component of target position (azimuth), relative to center of FOV *
y_t = y component of target position (elevation), relative to center of FOV *
v_x = x component of target velocity *
v_y = y component of target velocity *
a_x = x component of centroid acceleration (if included in model)
a_x = y component of centroid acceleration (if included in model)
x_a = x component of atmospheric jitter (1 or 3 states)
y_a = y component of atmospheric jitter (1 or 3 states)
x_p = offset (along velocity vector) of centroid from equilibrium point
v_p = velocity (along velocity vector) of pogo oscillation
x_o = offset between centroid equilibrium point and center-of-mass

* Note: The target is considered to be the missile center-of-mass for the elemental filter; it's the centroid for all others.

Each state in Equation (4-1) is coordinatized in the α-β (FLIR) plane of Section 2.3.1.4. For the observability gramian studies (the only models that included acceleration states) the target acceleration is represented as an exponentially time-correlated first order Gauss-Markov process [40]. A comparison between the filter model presented here and the fourteen-state truth model described earlier in Chapter III reveals the extent of state reduction. Note that the atmospheric jitter model can be reduced from the six states defined in the truth model in Chapter III to two states. The effect of the higher frequency double pole in each FLIR axis direction is negligible and is intentionally disregarded in the elemental filter to reduce the filter order [38]. Since the full six-state jitter model is the same as the one employed for the truth model, the two state approximated model will be presented here. Furthermore, the bending/vibration states defined in Section 3.2.3 are similarly excluded since past research found no significant degradation in filter performance without these states [17]. The pogo states defined in Section 3.2.4 are modeled identically to the truth model with the exception of scaling noise strengths for tuning purposes. Finally, the offset state was only included in the AFIT elemental filter. When it is included, the definition of the target dynamics states (but not the mathematical dynamics model) changes from centroid location to the center-of-mass location. As will be seen in Section 4.3, the measurement model does change slightly with this alteration.

The filter model is described by the time-invariant, linear stochastic differential equation [21]:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{F}\,\boldsymbol{x}(t) + \boldsymbol{G}\boldsymbol{w}(t) \tag{4-2}$$

where

F = time-invariant system (plant) matrix

$x(t)$ = n-dimensional filter state vector

G = n x m time-invariant noise distribution matrix

$w(t)$ = m-dimensional, white Gaussian noise process with independent components, and mean and covariance kernel statistics:

$$\begin{aligned} E\{w(t)\} &= 0 \\ E\left\{w(t)w^T(t+\tau)\right\} &= Q\,\delta(\tau) \end{aligned} \tag{4-3}$$

The filter state estimate and error covariance matrix are propagated forward to the next measurement update using the following discrete-time filter propagation equations [21]:

$$\hat{x}(t_{i+1}^-) = \Phi(\Delta t)\hat{x}(t_i^+) \tag{4-4}$$

$$P(t_{i+1}^-) = \Phi(\Delta t)P(t_i^+)\Phi^T(\Delta t) + Q_d \tag{4-5}$$

where

$\hat{x}(t_i)$ = filter estimate of the n-dimensional state vector

$\Phi(\Delta t)$ = n x n time-invariant state transition matrix for propagation over the sample period: $\Delta t = t_{i+1} - t_i$

$P(t_i)$ = n x n filter covariance matrix

(t_i^-) = time instant before measurement is incorporated into the estimate at time t_i

(t_i^+) = time instant after measurement is incorporated into the estimate at time t_i

Q_d = m x m filter dynamics discrete noise covariance given by:

$$Q_d = \int_{t_i}^{t_{i+1}} \Phi(t_{i+1}-\tau)GQG^T\Phi^T(t_{i+1}-\tau)\,d\tau \tag{4-6}$$

The components of Equations (4-2),(4-4) and (4-5) associated with each state are described in the next four sections.

4.2.1 Target Dynamics Model. Two separate dynamics models were used in this research. In the full six-state model the target acceleration is modeled as an exponentially time-correlated first order Gauss-Markov process. The acceleration model was not ever used in Monte Carlo analysis for this research. Since it was only used for the stochastic observability gramian models, and results were inconclusive (see Chapter V), it is not specified here. The model used for the observability gramian analysis was identical to the one detailed in Rizzo's thesis [40], so the interested reader is referred there for a full description.

The MSOFE and elemental filter models used a four-state model to describe target dynamics. This model is represents the velocity states as random constants plus noise [21]. The time-invariant continuous-time dynamics system matrix $\boldsymbol{F}_t$ is given by:

$$\boldsymbol{F}_t = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{4-7}$$

The noise distribution matrix $\boldsymbol{G}_t$ is:

$$\boldsymbol{G}_t = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{4-8}$$

The strength of the white Gaussian noise $\boldsymbol{w}_t$, given by $\boldsymbol{Q}_t$, is:

$$\boldsymbol{Q}_t = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{4-9}$$

The time invariant target dynamics state transition matrix $\Phi_t(\Delta t)$ is given by:

$$\Phi_t(\Delta t) = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4-10}$$

The filter dynamics discrete noise covariance Q_{dt} is given by:

$$Q_{dt} = \begin{bmatrix} \frac{1}{3}\Delta t^3 & 0 & \frac{1}{2}\Delta t^2 & 0 \\ 0 & \frac{1}{3}\Delta t^3 & 0 & \frac{1}{2}\Delta t^2 \\ \frac{1}{2}\Delta t^2 & 0 & \Delta t & 0 \\ 0 & \frac{1}{2}\Delta t^2 & 0 & \Delta t \end{bmatrix} \tag{4-11}$$

4.2.2 Atmospheric Disturbance Model. The atmospheric jitter model describes the motion of the plume image in the FLIR plane due to atmospheric disturbances (refraction variations from moisture, turbulence, thermal variation, etc.). The full six-state model developed by the Analytic Sciences Corporation is introduced in Section 3.2.2. This section presents an approximated model that implements a first-order Gauss Markov white noise process to simulate jitter along each FLIR plane axis. The time-invariant system matrix F_a of the continuous-time dynamics model of Equation (4-2) is:

$$F_a = \begin{bmatrix} -\omega_a & 0 \\ 0 & -\omega_a \end{bmatrix} \tag{4-12}$$

where

ω_a = atmospheric jitter break frequency, 14.14 rad/sec

The noise distribution matrix G_a is:

$$G_a = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{4-13}$$

The strength of the white Gaussian noise w_a, given by Q_a, is:

$$Q_a = \begin{bmatrix} \frac{2\sigma_a^2}{\tau_a} & 0 \\ 0 & \frac{2\sigma_a^2}{\tau_a} \end{bmatrix} \tag{4-14}$$

where

σ_a^2 = variance and mean-squared value for the atmospheric jitter process

τ_a = correlation time constant for atmospheric jitter process ($\tau_a = 1/\omega$)

The time invariant target dynamics state transition matrix $\Phi_a(\Delta t)$ is given by:

$$\Phi_a(\Delta t) = \begin{bmatrix} \exp(-\omega_a \Delta t) & 0 \\ 0 & \exp(-\omega_a \Delta t) \end{bmatrix} \tag{4-15}$$

The filter dynamics noise covariance Q_{da} is given by:

$$Q_{da} = \begin{bmatrix} q_{da11} & 0 \\ 0 & q_{da22} \end{bmatrix} \tag{4-16}$$

where

$$\begin{aligned} q_{da11} &= \sigma_a^2 \left[1 - \exp\left(-\frac{2(\Delta t)}{\tau_a} \right) \right] \\ q_{da22} &= \sigma_a^2 \left[1 - \exp\left(-\frac{2(\Delta t)}{\tau_a} \right) \right] \end{aligned} \tag{4-17}$$

In the MSOFE simulations, the atmospheric jitter variance and K_a (truth model gain adjustment from Section 3.2.2) were initially adjusted to yield a process noise strength in both the truth and filter models equivalent to $\sigma_a^2=0.2$. The dynamics noise parameter values for the filter were then adjusted during the tuning process.

4.2.3 Pogo Dynamics Model. The plume pogo dynamics model implemented in all filters was identical to the model described in Section 3.2.4. The time-invariant system matrix $\boldsymbol{F}_p$ is given by:

$$\boldsymbol{F}_p = \begin{bmatrix} 0 & 1 \\ -\omega_{pf}^2 & -2\zeta_{pf}\omega_{pf} \end{bmatrix} \tag{4-18}$$

where

ω_{pf} = undamped natural pogo frequency (0.1 - 10 Hz, 1 Hz nominal)

ς_{pf} = filter damping coefficient chosen to be 0.05 [40]

The noise distribution matrix $\boldsymbol{G}_p$ is:

$$\boldsymbol{G}_p = \begin{bmatrix} 0 \\ K_{pf}\omega_{pf}^2 \end{bmatrix} \tag{4-19}$$

and the white Gaussian noise w_p, given by Q_p, is of unit strength:

$$Q_p = 1 \tag{4-20}$$

The time invariant target dynamics state transition matrix $\Phi_p(\Delta t)$ is given by:

$$\Phi_p(\Delta t) = \begin{bmatrix} \Phi_{p11}(\Delta t) & \Phi_{p12}(\Delta t) \\ \Phi_{p21}(\Delta t) & \Phi_{p22}(\Delta t) \end{bmatrix} \tag{4-21}$$

and the scalar elements of $\Phi_p(\Delta t)$ are the same as given in Equation (3-43), Section 3.2.4. The filter dynamics noise covariance $\boldsymbol{Q}_{da}$ is not included here due to its extreme length and the complexity of its elements. The interested reader is referred to the software for a complete description of this matrix.

4.2.4 Centroid Equilibrium Point/Center-of-Mass Offset Model. In previous theses [8,9,14], the measurement determined from the LEL (by either speckle return or Doppler spectra of the plume and hardbody) was processed in an independent center-of-mass offset filter. The estimate from that filter was then added to a FLIR filter that estimated the position of the centroid in order to obtain center-of-mass position. Since pogo was not included in the filter models, the offset between the center-of-mass and the intensity centroid was modeled as a constant. For this thesis, the same dynamics model is used, but the offset state is augmented to the previous models to form a single nine-state AFIT elemental filter. The constant offset state describes the distance between the equilibrium point about which the plume pogos. The offset is modeled as a bias. Equations (4-2), (4-4), and (4-5) are still applicable but are expressed in scalar form since there is only a single state. The bias is modeled as a simple integrator, with driving pseudo-noise for filter tuning purposes.

The elements of the linear, time-invariant stochastic differential equation are:

F_o = 0

G_o = time-invariant noise distribution matrix, equal to unity

$w_o(t)$ = white Gaussian noise process, independent of the noises driving the target dynamics and atmospheric jitter models, with mean and covariance kernel statistics:

$$\begin{aligned} E\{w_o(t)\} &= 0 \\ E\{w_o(t)w_o(t+\tau)\} &= Q_o\,\delta(\tau) \end{aligned} \tag{4-22}$$

and $Q_o = 1$.

The elements of the equivalent discrete-time filter propagation Equations (4-4) and (4-5) are given by:

$\Phi_o(\Delta t)$ = time-invariant state transition matrix, equal to unity

Q_{do} = filter dynamics noise variance given by equal to $Q_o \Delta t$

4.3 Measurement Models

This section discusses the filter measurement models implemented for this thesis. The first subsection explains how the raw FLIR data is processed through an enhanced correlation algorithm. By use of templates, the algorithm generates "pseudo-measurements" for the linear Kalman filter. The linear and non-linear update functions employed for the Kalman filters are also presented. Finally, the last subsection describes the measurement model used in conjunction with the low-energy laser measurements.

4.3.1 FLIR Measurement Model. Measurements of the intensity centroid's position are generated by an enhanced correlator algorithm, shown in Figure 4.1, developed by Rogers [27,41]. Unlike the standard correlator tracker that correlates the current FLIR data frame with the previous data frame, this enhanced correlator algorithm correlates the current FLIR data frame with a template that represents an estimate of the target plume's intensity function. "Psuedo-measurements" of the centroid's position offsets are produced by this correlator, and since the "pseudo-measurements" are of a nonharsh nonlinearity with respect to the states being estimated, this tracker uses a nearly linear Kalman filter. Because the enhanced correlator algorithm and the template generation have not changed, the next two sections describing their operation are reproduced from the previous thesis with slight modifications [14].

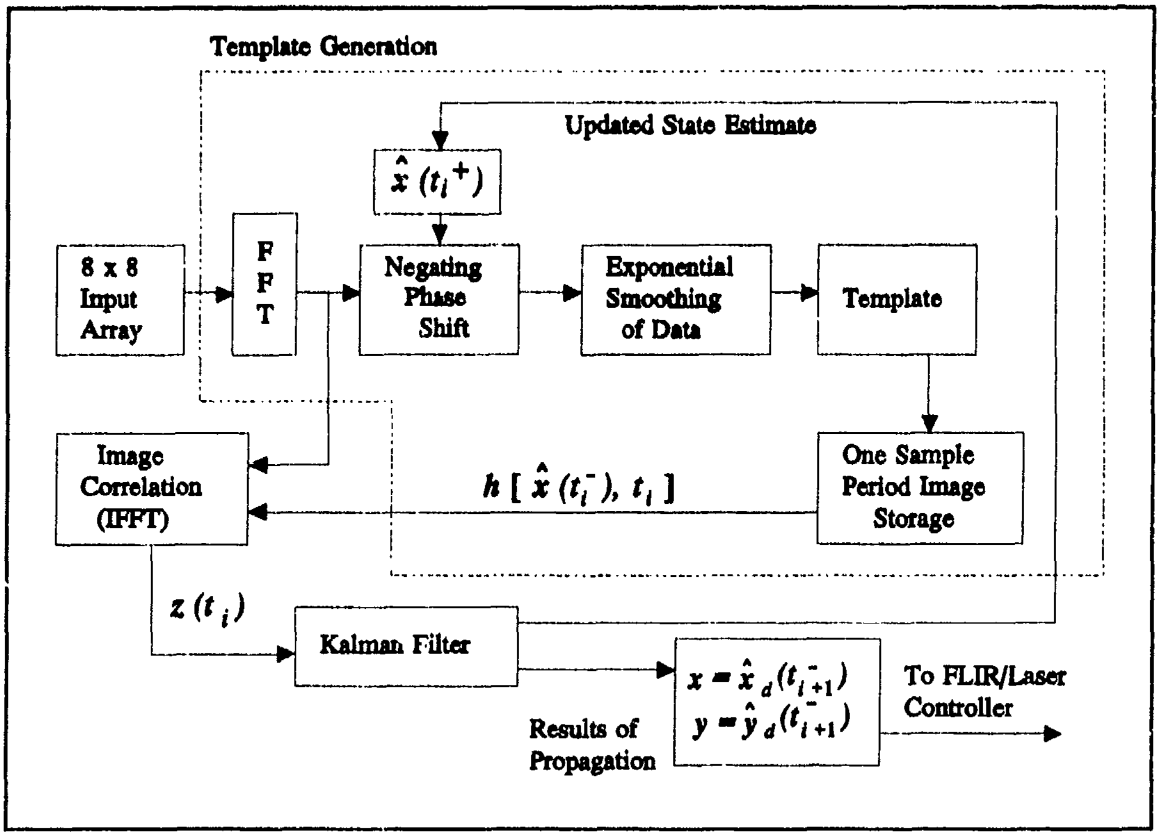


Figure 4.1 Linear Kalman Filter/Enhanced Correlator Algorithm

4.3.1.1 Enhanced Correlator Algorithm. The algorithm presented here was developed as an alternative to an earlier 64-dimensional, nonlinear measurement model. Previously, an extended Kalman filter processed raw FLIR measurement data from a standard FLIR sensor, with no correlation algorithm utilized at all [31]. With the enhanced correlator algorithm, a nearly linear Kalman filter is employed since the output measurements from the correlation algorithm are 2-dimensional position measurements that are nonharsh nonlinear functions of the states to be estimated. As will be seen in later in the section, the nonlinearity is a sinusoidal function caused by the introduction of the plume pogo effect into the model. The sinusoidal function is a mild nonlinearity in comparison the extended Kalman filter model required to process raw FLIR data. This enhanced correlator/linear Kalman filter configuration performed as well as the extended Kalman filter with respect to rms tracking errors and further provided a reduction in computational loading. The "enhancement" occurs in the following manner [41]:

1. The most current FLIR data is correlated with a template (which is an estimate of the target's intensity function), instead of with the previous FLIR data frame.

2. Instead of outputting the peak of the correlation function, a technique known as "thresholding" is used along with a simple center-of-mass computation. The enhanced correlator outputs the center-of-mass of the portion of the correlation function that is greater than some predetermined lower bound. Consequently, the enhanced correlator has no difficulty distinguishing global peaks from local peaks, as do many conventional "peak-finding" correlation algorithms.

3. The FLIR/laser pointing commands are generated via the Kalman filter propagation cycle instead of by the "raw measurement" output of a standard correlation algorithm.

5. The Kalman filter estimate, $\hat{x}(t_i^+)$, is used to center the template, so that the offsets seen in the enhanced correlator algorithm should be smaller than those visible in the conventional correlator. This increases the amount of "overlap" between the actual FLIR data and the stored template, and thus improves performance.

Referring back to Figure 4.1, the enhanced correlation algorithm uses the 8 x 8 array of target intensities obtained by the FLIR measurement, to establish a 64-element shape function from the target plume intensity profile (Section 3.3.1). The intensity functions are centered on the FLIR plane by translational shifts using centroid offset estimates from $\hat{x}(t_i^+)$, using the "shifting property" of the Fourier Transform, where negating phase shifts are applied in the spatial frequency domain to accomplish a translational shift in the original domain. Rather than perform the difficult correlation in the time domain, the Fourier domain allows one to apply straightforward multiplication to implement the "translational shift" of the intensity functions and eventual correlation with the template. Exponential smoothing is then used to average the result with previously centered images to yield an updated template. The current FLIR data is then correlated against the template of the previously stored shape function that has been centered on the FLIR image plane. The outputs of the algorithm are two linear offsets, x_c and y_c as shown in Equations (3-1) and (3-2), that yield the highest correlation of the current data with the template. These "pseudo-measurements" are then fed to the linear FLIR Kalman filter for its update. The filter provides the updated estimate, $\hat{x}(t_i^+)$, used to center the FLIR intensity profile to be included in the template generation for the next measurement.

4.3.1.2 Template Generation. The template reconstructs the shape, size, and location of the intensity centroid using the raw noise-corrupted FLIR measurements. [illegible] template generation begins with an input of a FLIR frame of data to the enhanced correlator algorithm of Figure 4.1. Using the "shifting" property of the fast Fourier transform (FFT), which states that a translational shift in the spatial domain is equivalent to a linear phase shift in the frequency domain, the required phase shift is computed by:

$$F\{g(x - x_{shift}, y - y_{shift})\} = G(f_x, f_y)\exp\{-j2\pi(f_x \cdot x_{shift} + f_y \cdot y_{shift})\} \qquad (4\text{-}23)$$

where

$F\{\cdot\}$ = Fourier transform operator
$g(x,y)$ = 2-dimensional spatial data array
$G(f_x, f_y)$ = $F\{g(x,y)\}$
f_x, f_y = spatial frequencies

The Fourier transform is implemented in the simulation software using the Cooley-Tukey algorithm [41]. The target plume intensity shape function is "centered on the FLIR plane" by phase shifting the transformed function an amount equal to:

$$\begin{aligned} x_{shift}(t_i) &= \hat{x}_t(t_i^+) + \hat{x}_a(t_i^+) + \hat{x}_p(t_i^+)\cos\theta_f \\ y_{shift}(t_i) &= \hat{y}_t(t_i^+) + \hat{y}_a(t_i^+) - \hat{x}_p(t_i^+)\sin\theta_f \end{aligned} \tag{4-24}$$

where $\hat{x}_t$, $\hat{y}_t$, $\hat{x}_a$, $\hat{y}_a$, and $\hat{x}_p$ are the state estimates defined in Equation (4-1), and θ_f is the filter's estimate of the velocity vector angle in the FLIR plane. For the phase shifting function, the target dynamics states (denoted by subscript t), *always* refer to the centroid center-of-intensity. In the AFIT elemental filter, the target dynamics states of Equation (4-1), refer to the hardbody center-of-mass, so the estimates used in Equation (4-24) are calculated by subtracting the filter's estimated offset from those states to obtain $\hat{x}_t$ and $\hat{y}_t$. Once the data is centered on the FLIR plane, it is incorporated into an updated template for the next sample period. In the simulation, the Kalman filter's first update cycle is bypassed to form the initial template.

The template is generated by averaging the N most recent centered intensity functions observed by the FLIR sensor. The averaging process tends to accentuate the target intensity function and attenuate the corrupting background and FLIR noises. The memory size N is chosen according to how rapidly the shape functions change, i.e., highly dynamic intensity functions require small values of N, while slowly varying functions use large N values. Typically, a true finite memory averager would require a large computer memory [22]. However, the enhanced correlator algorithm circumvents the memory storage issue by incorporating an "exponential smoothing" technique to approximate the averaging. This technique has properties similar to finite memory averaging, but with the advantage of requiring only the storage of a single FLIR frame of data. The template is maintained by the exponential smoothing algorithm given by:

$$\hat{I}(t_i) = \gamma I(t_i) + (1 - \gamma)\hat{I}(t_{i-1}) \tag{4-25}$$

where

$\hat{I}(t_i)$ = "smoothed estimate" (template) of the target's intensity function
$I(t_i)$ = "raw" intensity function from the current FLIR data frame
γ = smoothing constant: $0 < \gamma \leq 1$

The smoothing constant γ is comparable to the value selected for N. From Equation (4-25), it can be seen that large values of γ emphasize the current data frame and correspond to small values of N. Based on previous studies [19,45], a smoothing constant of $\gamma = 0.1$ is used for this thesis.

A reinitialization algorithm is used once after the first ten sample periods (although it could be called periodically thereafter as well, in actual implementation). Once the template is computed, its centroid is calculated and shifted to the center of field of view for the template, thus eliminating any biases. It is this template which is now stored and correlated with the next FLIR data to produce the "pseudo-measurements."

4.3.1.3 "Pseudo-Measurements". The template serves as the best estimate of the shape of the target plume intensity function prior to receiving a new FLIR data frame. The cross-correlation of the incoming FLIR data with the template provides the position offsets from the center of the FOV to the centroid of the target intensity image. The cross-correlation is computed by taking the inverse fast Fourier transform (IFFT) of the equation [41]:

$$F\{g(x,y) * l(x,y)\} = G(f_x, f_y)L^*(f_x, f_y) \tag{4-26}$$

where

$F\{\cdot\}$ = Fourier transform operator
$g(x,y)$ = measured target intensity function of the current FLIR data frame
$l(x,y)$ = expected target plume intensity function (i.e., template)
$g(x,y) * l(x,y)$ = cross-correlation of $g(x,y)$ and $l(x,y)$
$G(f_x, f_y)$ = $F\{g(x,y)\}$
$L^*(f_x, f_y)$ = complex conjugate of $F\{l(x,y)\}$

After the IFFT is accomplished, the values of the correlation function, $g(x,y) * l(x,y)$, are modified such that any value less than 0.3 of the function's maximum value is set to zero [19,37]. This "thresholding" technique is used to eliminate false peaks in the correlation function that occur due to noise and other effects. As shown earlier in Figure 4.1, the output of the image correlation is the offset of the "thresholded" FLIR intensity centroid from the center of the FLIR FOV. This offset is assumed to be the result of the summed effects of target dynamics, atmospheric jitter, the pogo effect, and measurement noise.

If the filter only receives FLIR measurements and therefore only tracks the plume centroid (i.e. no LEL measurements), as was the case with the "FLIR filter" used in previous theses [8,9,14] and the filter implemented in MSOFE for this thesis, then the x- and y-components of the offsets are the pseudo-measurements provided to the Kalman filter. In this case the target dynamic states describe the motion of the centroid. For the following development, an eight-state filter used in MSOFE simulations will serve as an example. The state vector $\mathbf{x}$ of Equation (4-1) consists of 4 target dynamics, 2 atmospheric jitter (one state per axis) and 2 pogo states.

The offsets from the enhanced correlator are expressed in terms of Equation (4-1) as:

$$\begin{aligned} x_{offset} &= x_t + x_a + x_p \cos\theta_f + v_{f1} \\ y_{offset} &= y_t + y_a - x_p \sin\theta_f + v_{f2} \end{aligned} \qquad (4\text{-}27)$$

where

$$\begin{aligned} \cos\theta_f &= \frac{v_x}{\sqrt{v_x^2 + v_y^2}} \\ \sin\theta_f &= \frac{-v_y}{\sqrt{v_x^2 + v_y^2}} \end{aligned} \qquad (4\text{-}28)$$

and θ_f is the angle between the velocity vector and the x-axis of the FLIR plane. These two measurements can be represented in state form as:

$$z(t_i) = \boldsymbol{h}_f[\boldsymbol{x}_f(t_i), t_i] + \boldsymbol{v}_f(t_i) \qquad (4\text{-}29)$$

where

$z(t_i)$ = $[\, x_{offset}(t_i),\ y_{offset}(t_i)\,]^T$; measured in pixels

$\boldsymbol{h}_f[\boldsymbol{x}_f(t_i), t_i]$ = nonlinear measurement function vector given by Equation (4-27)

$\boldsymbol{x}_f(t_i)$ = state vector of Equation (4-1)

$\boldsymbol{v}_f(t_i)$ = 2-dimensional, discrete-time, white Gaussian measurement noise (in pixels) with statistics:

$$\begin{aligned} E\{v(t_i)\} &= 0 \\ E\left\{v(t_i)v^T(t_j)\right\} &= \begin{Bmatrix} R & t_i = t_j \\ 0 & t_i \neq t_j \end{Bmatrix} \end{aligned} \qquad (4\text{-}30)$$

Note that because of the pogo states being defined along the velocity vector and being included in the output equations, this measurement model is nonlinear in the filter states and the extended

Kalman filter update cycle described in Chapter II (Equations (2-26) and (2-27)) must be applied. The equations are [21]:

$$\begin{aligned} \boldsymbol{K}(t_i) &= \boldsymbol{P}_f(t_i^-)\boldsymbol{H}_f^T[\boldsymbol{H}_f\boldsymbol{P}_f(t_i^-)\boldsymbol{H}_f^T + \boldsymbol{R}_f]^{-1} \\ \hat{\boldsymbol{x}}_f(t_i^+) &= \hat{\boldsymbol{x}}_f(t_i^-) + \boldsymbol{K}(t_i)\left\{\boldsymbol{z}(t_i) - \boldsymbol{h}_f[\hat{\boldsymbol{x}}_f(t_i^-), t_i]\right\} \\ \boldsymbol{P}_f(t_i^+) &= \boldsymbol{P}_f(t_i^-) - \boldsymbol{K}(t_i)\boldsymbol{H}_f\boldsymbol{P}_f(t_i^-) \end{aligned} \tag{4-31}$$

where

$\boldsymbol{K}(t_i)$ = 8 x 2 filter gain matrix

$\boldsymbol{P}_f(t_i)$ = 8 x 8 filter covariance matrix

$\boldsymbol{h}_f[\boldsymbol{x}_f(t_i),t_i]$ = 2-dimensional non-linear measurement function; Equation (4-29)

$\boldsymbol{H}_f$ = linearized measurement matrix; Equation (2-26)

$\boldsymbol{R}_f$ = 2 x 2 measurement noise covariance matrix; Equation (4-34)

$\hat{\boldsymbol{x}}_f(t_i)$ = 8-dimensional estimated state vector; Equation (4-1)

$\boldsymbol{z}(t_i)$ = 2-dimensional measurement vector; Equation (4-29)

(t_i^-) = time instant immediately before measurements are incorporated at time t_i

(t_i^+) = time instant immediately after measurements are incorporated at time t_i

For the example 8-state filter, using Equations (4-27) and (4-28), the linearized FLIR measurement matrix $\boldsymbol{H}_f$ is given by:

$$\boldsymbol{H}_f = \begin{bmatrix} 1 & 0 & H_{13} & H_{14} & 1 & 0 & H_{17} & 0 \\ 0 & 1 & H_{23} & H_{33} & 0 & 1 & H_{27} & 0 \end{bmatrix} \tag{4-32}$$

where the scalar elements are:

$$H_{13} = \frac{\partial h_1[\boldsymbol{x}, t_i]}{\partial x_3} = \left. \frac{x_7 x_4^2}{\sqrt[\frac{3}{2}]{x_3^2 + x_4^2}} \right|_{\boldsymbol{x} = \hat{\boldsymbol{x}}_f(t_i^-)} \tag{4-33a}$$

$$H_{14} = \frac{\partial h_1[\boldsymbol{x}, t_i]}{\partial x_4} = \left. \frac{-x_7 x_3 x_4}{\sqrt[\frac{3}{2}]{x_3^2 + x_4^2}} \right|_{\boldsymbol{x} = \hat{\boldsymbol{x}}_f(t_i^-)} \qquad (4\text{-}33b)$$

$$H_{17} = \frac{\partial h_1[\boldsymbol{x}, t_i]}{\partial x_7} = \left. \frac{x_3}{\sqrt{x_3^2 + x_4^2}} \right|_{\boldsymbol{x} = \hat{\boldsymbol{x}}_f(t_i^-)} \qquad (4\text{-}33c)$$

$$H_{23} = \frac{\partial h_2[\boldsymbol{x}, t_i]}{\partial x_3} = \left. \frac{-x_7 x_3 x_4}{\sqrt[\frac{3}{2}]{x_3^2 + x_4^2}} \right|_{\boldsymbol{x} = \hat{\boldsymbol{x}}_f(t_i^-)} \qquad (4\text{-}33d)$$

$$H_{24} = \frac{\partial h_2[\boldsymbol{x}, t_i]}{\partial x_4} = \left. \frac{x_7 x_3^2}{\sqrt[\frac{3}{2}]{x_3^2 + x_4^2}} \right|_{\boldsymbol{x} = \hat{\boldsymbol{x}}_f(t_i^-)} \qquad (4\text{-}33e)$$

$$H_{27} = \frac{\partial h_2[\boldsymbol{x}, t_i]}{\partial x_7} = \left. \frac{x_4}{\sqrt{x_3^2 + x_4^2}} \right|_{\boldsymbol{x} = \hat{\boldsymbol{x}}_f(t_i^-)} \qquad (4\text{-}33f)$$

Note that, in the observability tests for which θ_f was artificially assumed to be known, the H_{13}, H_{14}, H_{23}, and H_{24} terms in Equation (4-31) become zero and $H_{17} = \cos\theta_f$ and $H_{27} = -\sin\theta_f$.

The measurement noise $v_f(t_i)$ represents the combined corrupting effects of the spatially correlated background noise, the FLIR sensor noise (Section 4.3.1), and the errors due to the FFT/IFFT processes. The covariance matrix $\boldsymbol{R}_f$ (with units of pixels^2) associated with this error is given by [13,33,41]:

$$\boldsymbol{R}_f = \begin{bmatrix} 0.00436 & 0 \\ 0 & 0.00436 \end{bmatrix} \qquad (4\text{-}34)$$

For the elemental filter implemented in the AFIT software, a ninth state, centroid equilibrium point to center-of-mass offset x_o, is augmented to the state vector of Equation (4-1). The four target dynamics states (x_t, y_t, v_x, v_y) in this case describe the motion of the hardbody center-of-mass. The fundamental extended Kalman filter update equations remain the same, but the non-linear measurement function and associated partials take on a new meaning. Equations (4-27) become:

$$\begin{aligned} x_{offset} &= x_t + x_a + (x_p - x_o)\cos\theta_f + v_{f1} \\ y_{offset} &= y_t + y_a - (x_p - x_o)\sin\theta_f + v_{f2} \end{aligned} \tag{4-35}$$

and $\cos\theta_f$ and $\sin\theta_f$ retain the same meaning given in Equation (4-28).

The linearized measurement matrix $\boldsymbol{H}_f$ is given by:

$$\boldsymbol{H}_f = \begin{bmatrix} 1 & 0 & H_{13} & H_{14} & 1 & 0 & H_{17} & 0 & H_{19} \\ 0 & 1 & H_{23} & H_{33} & 0 & 1 & H_{27} & 0 & H_{29} \end{bmatrix} \tag{4-36}$$

The new partials of $h[\cdot,\cdot]$ are given by:

$$H_{13} = \frac{\partial h_1[\boldsymbol{x}, t_i]}{\partial x_3} = \left. \frac{x_4^2(x_7 - x_9)}{{}^{\frac{3}{2}}\!\sqrt{x_3^2 + x_4^2}} \right|_{\boldsymbol{x} = \hat{\boldsymbol{x}}_f(t_i^-)} \tag{4-37a}$$

$$H_{14} = \frac{\partial h_1[\boldsymbol{x}, t_i]}{\partial x_4} = \left. \frac{-x_3 x_4(x_7 - x_9)}{{}^{\frac{3}{2}}\!\sqrt{x_3^2 + x_4^2}} \right|_{\boldsymbol{x} = \hat{\boldsymbol{x}}_f(t_i^-)} \tag{4-37b}$$

$$H_{17} = \frac{\partial h_1[\boldsymbol{x}, t_i]}{\partial x_7} = \left. \frac{x_3}{\sqrt{x_3^2 + x_4^2}} \right|_{\boldsymbol{x} = \hat{\boldsymbol{x}}_f(t_i^-)} \tag{4-37c}$$

$$H_{19} = \frac{\partial h_1[\boldsymbol{x}, t_i]}{\partial x_9} = \left. \frac{-x_3}{\sqrt{x_3^2 + x_4^2}} \right|_{\boldsymbol{x} = \hat{\boldsymbol{x}}_f(t_i^-)} \tag{4-37d}$$

$$H_{23} = \frac{\partial h_2[\boldsymbol{x}, t_i]}{\partial x_3} = \left. \frac{-x_3 x_4 (x_7 - x_9)}{\sqrt[\frac{3}{2}]{x_3^2 + x_4^2}} \right|_{\boldsymbol{x} = \hat{\boldsymbol{x}}_f(t_i^-)} \tag{4-37e}$$

$$H_{24} = \frac{\partial h_2[\boldsymbol{x}, t_i]}{\partial x_4} = \left. \frac{x_3^2 (x_7 - x_9)}{\sqrt[\frac{3}{2}]{x_3^2 + x_4^2}} \right|_{\boldsymbol{x} = \hat{\boldsymbol{x}}_f(t_i^-)} \tag{4-37f}$$

$$H_{27} = \frac{\partial h_2[\boldsymbol{x}, t_i]}{\partial x_7} = \left. \frac{x_4}{\sqrt{x_3^2 + x_4^2}} \right|_{\boldsymbol{x} = \hat{\boldsymbol{x}}_f(t_i^-)} \tag{4-37g}$$

$$H_{29} = \frac{\partial h_2[\boldsymbol{x}, t_i]}{\partial x_9} = \left. \frac{-x_4}{\sqrt{x_3^2 + x_4^2}} \right|_{\boldsymbol{x} = \hat{\boldsymbol{x}}_f(t_i^-)} \tag{4-37h}$$

The state dimension is now nine, so the dimensions of the matrices in Equation (4-31) change accordingly. The reader should keep in mind that only two FLIR measurements are brought in at this update, so the measurement dimension remains at two, therefore, the corresponding measurement noise covariance matrix $\boldsymbol{R}_f$ remains the same as in the eight-state measurement update developed earlier.

4.3.2 Doppler Measurement Model. The primary objective of this research is the precise tracking of the missile hardbody and determination of its center-of-mass location in the presence of plume pogo. The basic premise underlying the dynamics modeling efforts is that the center-of-mass is located at a constant offset distance relative to an equilibrium point about which the intensity centroid oscillates. The offset distance is oriented angularly using the filter-estimated velocity [illegible] in the FLIR image plane [8]. Figure 4.2 illustrates the geometry of estimating the offset distance and the dependence of the scan and offset computation upon the filter's estimates of the position and velocity of the intensity centroid immediately after the FLIR update. (Note that Figure 4.2 depicts the ideal situation; in general, the filter estimates of the centroid position, velocity, and the orientation angle are not equal to the truth model values.) Originating at the intensity centroid's estimated position, a low-energy laser is scanned along the estimated velocity vector. Ideally, identification of reflections from the hardbody allow determination of the near

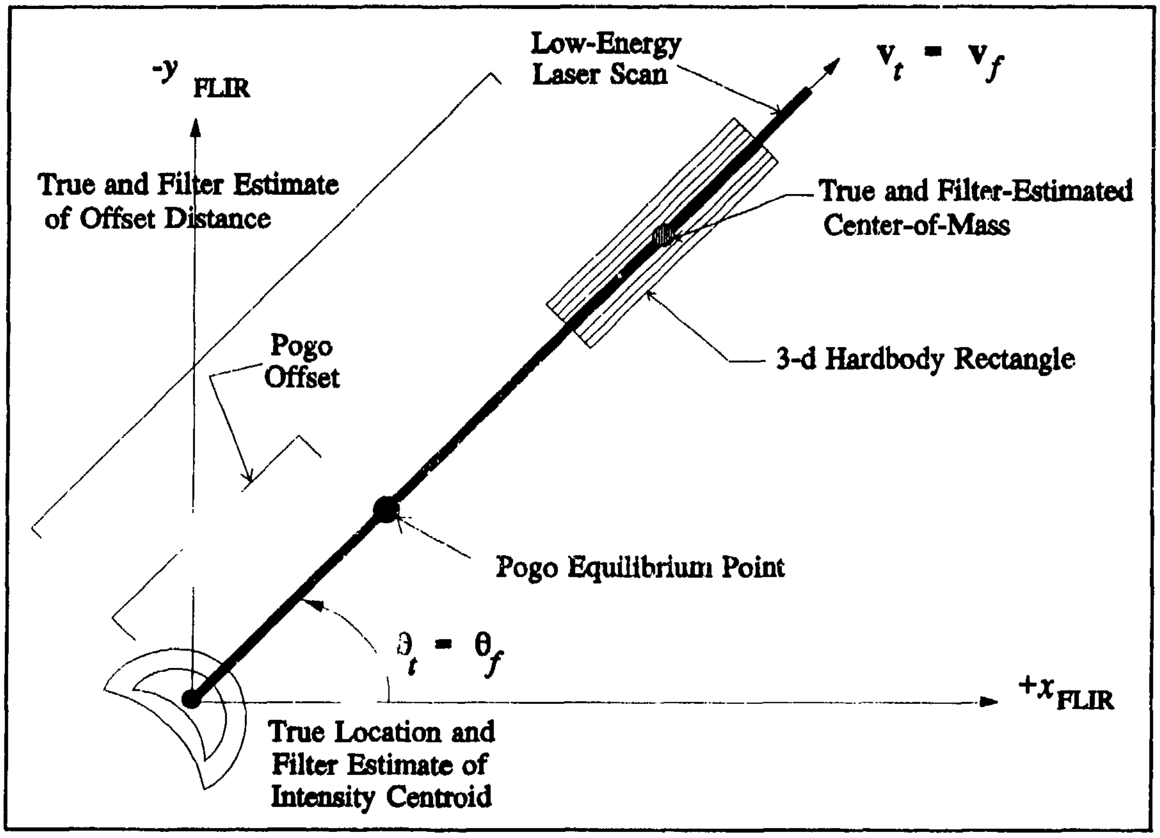

$-y_{FLIR}$
Low-Energy Laser Scan
$v_t = v_f$
True and Filter Estimate of Offset Distance
True and Filter-Estimated Center-of-Mass
3-d Hardbody Rectangle
Pogo Offset
Pogo Equilibrium Point
$\theta_t = \theta_f$
$+x_{FLIR}$
True Location and Filter Estimate of Intensity Centroid

Figure 4.2 Filter Estimate of Offset Distance (Ideal Conditions)

and far end of the missile, so the center-of-mass is calculated as the midpoint between the two ends. The reflections of the low-energy scan generate a measurement of the offset distance, to be utilized as aimpoint information for the high-energy laser. The offset measurement delivered by the scan is a function of the constant offset plus translation of the centroid from its equilibrium point due to the pogo phenomenon. Because the offset was only modeled in the AFIT elemental filter, the Doppler offset measurement is only applicable to the AFIT software simulations, which used the nine-state filter.

The previous theses [8,9,14] utilized a two-filter approach in which the FLIR filter and center-of-mass filter functioned autonomously; the FLIR filter had no knowledge of the existence of the center-of-mass filter. Both Eden [8] and Evans [9] utilized the low-energy-laser speckle return of the hardbody/plume interface to generate measurements for the center-of-mass filter. Herrera [14] utilized the Doppler spectra in the laser return to derive successfully a more accurate, unbiased, offset measurement, but still maintained the same basic independent-filter structure. Since the Doppler measurement model used for this research was developed in the previous thesis [14], the description in this section has been reproduced with some revisions.

The Doppler measurement model provides a measurement based upon the low-energy laser Doppler return of the hardbody. The significant dissimilarities between the plume and hardbody-induced Doppler returns can be exploited to discern the plume/hardbody interface (Section 3.3.3) precisely, and provide information regarding the location of the hardbody. The low-energy laser measurement is provided to the Kalman filter whenever the laser intercepts the hardbody, and the hardbody-induced (and plume-induced) Doppler return is received by Doppler return sensor equipment. The resulting measurement to be provided to the filter is a noise-corrupted offset distance which is a linear function of filter's offset and pogo estimates. The measurement function is given by:

$$x_{offset} = x_9 - x_7 \tag{4-40}$$

The discrete-time scalar measurement model is given by:

$$z(t_i) = H_o x_o(t_i) + v_f(t_i) \tag{4-41}$$

where

$z(t_i)$ = measurement of the offset distance

H_o = measurement matrix

$x_o(t_i)$ = center-of-mass offset state

$v_f(t_i)$ = discrete-time, white Gaussian measurement noise with statistics:

$$\begin{aligned} E\{v(t_i)\} &= 0 \\ E\{v(t_i)v(t_j)\} &= \begin{Bmatrix} R_f & t_i = t_j \\ 0 & t_i \neq t_j \end{Bmatrix} \end{aligned} \tag{4-42}$$

where $R_f = R_t$ (true Doppler measurement variance), a function of low-energy laser wavelength, radar aperture, and signal-to-noise ratio (Section 3.3.3). The measurement matrix H_o is defined as:

$$H_o = [\ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ -1\ \ 0\ \ 1\ ] \tag{4-43}$$

Since the measurement is linear, the linear Kalman filter update cycle described in Section 2.2.2 (Equations (2-22)-(2-24)) is used.

In some instances, the low-energy laser sweep may be unsuccessful in generating a measurement, either due to missing the hardbody because of poor estimation of the centroid location and velocity vector orientation, or due to poor conditions for discerning the differences in Doppler frequency spectra between the hardbody and plume. If a measurement is not generated bye the LEL sweep, the LEL update is skipped.

4.3.3 Filter Parameters. This section provides a consolidated reference of the parameters used in the AFIT simulation. Presented below are definitions of the modeling parameters, initial conditions, and tuning parameters for the nine-state filter employed in this research.

4.3.3.1 Initial Conditions. Since initial acquisition characteristics of the filter have been explored in the past [47], emphasis is placed upon the tracking problem, rather than acquisition and tracking. Thus, taken from previous research [8,9,14], the filter initial state estimate $\hat{x}_o$ is artificially initialized to zero error for the position, velocity, and pogo states of Equation (4-1). The position states x_1 and x_2 are initialized on the true center-of-mass with the target plume intensity centroid centered in the FLIR FOV. The velocity states x_3 and x_4 are initialized in accordance with the target's initial trajectory conditions as defined in Section 3.4.1. Both atmospheric states x_5 and x_6 are initialized to zero. The offset of the plume from its equilibrium point is initialized to zero, as well as the velocity of the oscillation. The constant distance between the equilibrium point and the center-of-mass are also initialized to true conditions (87.5 meters, or 2.92 pixels).

The initial state covariance matrix $P(t_o)$ is:

$$P(t_o) = \begin{bmatrix} 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2000 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2000 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & .2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & .2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & .2 \end{bmatrix} \tag{4-44}$$

where the units of the covariance associated with the position states x_1 and x_2, the atmospheric states x_5 and x_6, the pogo position state x_p, and the offset state x_o, are pixels^2, and those of the velocity states x_3 and x_4 and the pogo velocity state v_p are expressed in pixels^2/seconds^2 [9].

4.3.3.2 Tuning Values. The measurement covariance matrix for the FLIR $\boldsymbol{R}_{FLIR}$ was established empirically in past research [27,41]. $\boldsymbol{R}_{FLIR}$ (with units of pixels^2) is given by:

$$\boldsymbol{R}_{FLIR} = \begin{bmatrix} 0.00363 & 0 \\ 0 & 0.00598 \end{bmatrix} \tag{4-45}$$

The measurement variance for the Doppler measurement $R_{Doppler}$, is equal to the true measurement variance and is a function of the low-energy laser wavelength, SNR, and aperture diameter of the transmitter. The filter measurement variance is carried over from Herrera's research and is given by [14]:

$$R_{Doppler} = R_t = \left[\frac{\theta_B}{3k_p\sqrt{\mathrm{SNR}}} \right]^2 \tag{4-46}$$

where

$R_{Doppler}$ = filter measurement variance

R_t = true measurement variance

θ_B = beam diffraction limit

k_p = pixel proportionality constant, 15 μrads/pixel

SNR = signal-to-noise ratio; 10

Both atmospheric variances σ_{ax}^2 and σ_{ay}^2 are equal to 0.2 pixels^2 in accordance with the truth model (Section 3.2.2)[9]. The process noise strength for the pogo state is dependent upon the desired rms amplitude of the pogo oscillation σ_p^2 [40]. The filter pogo gain constant K_{pf} is initially set equal to the truth pogo gain constant K_p (Section 3.2.4, Equation (3-38)) and then adjusted if necessary while leaving the truth noise strength constant. Nominal rms pogo amplitude for this research was carried over from Rizzo's research and set to $\sigma_p^2 = .112$ pixels^2 at a frequency of $\omega_p = 1$ Hz. The offset state dynamics noise variance Q_{do} is equal to 0.7 pixels^2, based upon Evans' research [9]. The probability of miss for the Doppler measurement model was set at 0.01 (see Section 3.3.3.4).

4.4 Summary

This chapter presented the dynamics models associated with the states incorporated into the various filters tested in this thesis. The dynamics model was presented in this fashion to enable the reader to determine the distinct model used for each filter discussed in the next chapter. The measurement models for both the FLIR and Doppler measurements were also introduced. The Kalman filter accepts pseudo-measurements generated by the enhanced correlator algorithm after processing the raw FLIR data. The update is implemented through use of the extended Kalman filter update cycle, since the pseudo-measurements are nonlinear in the state vector.

Measurement updates for both the MSOFE filter models and the AFIT model were discussed. Section 4.3.2 explained how the AFIT filter performs an additional linear update when the Doppler return measurement is made available. Finally, a summary of the initial conditions and tuning parameters was presented.

V. Observability Gramian and MSOFE Analysis

5.1 Introduction

This chapter presents the procedures followed for performing the observability studies referenced in the objectives in Chapter I. The rationale and a brief theoretical description of the analysis methods used are discussed in addition to the results obtained. The next section of this chapter enumerates the filters simulated and analyzed in MSOFE. The structure of each filter is introduced and a justification for its analysis is provided. Finally, the results of the Monte Carlo simulations in MSOFE will be presented for each filter configuration. The conclusions resulting from the stochastic observability tests and MSOFE Monte Carlo analysis will be presented in this chapter and summarized again in Chapter VII, along with the associated recommendations arising from the analyses.

None of the filter structures analyzed in this chapter include the LEL measurements. Since the principal intent of both the stochastic observability tests and the MSOFE analysis was to assist in verifying some of the difficulties Rizzo identified, the same types of models were used (and none of Rizzo's models included LEL measurements). Therefore, the offset state is not included in any of the models. In these instances, one would expect to find observability problems (especially on offset and pogo and possibly jitter). On the other hand, inclusion of the third measurement, as in the elemental filter discussed in the next chapter, lends insight directly into the offset and pogo states and one would expect observability to improve in the offset, pogo, and possibly the jitter states. An equivalent set of observability tests would have to be accomplished on the augmented filter structure with the third measurement to verify any improvement.

5.2 Stochastic Observability Tests

This section will present the procedures followed for performing the stochastic observability analyses. First, a brief review of why these studies were initiated is outlined. Next, a brief explanation of the mathematical theory is included, followed by a description of the filters analyzed and the results obtained. Conclusions from the findings complete this section.

5.2.1 Background Review. During the course of his research, Rizzo discovered a number of estimation problems with the simulations in the AFIT software [40]. He tested two different versions of the elemental filter. The first included eight states: six target dynamics states (position, velocity, acceleration along the FLIR plane axes), and two atmospheric jitter states. His second filter included the same eight states plus the two-state pogo model used in this thesis. He found that, contrary to expectations, inclusion of the pogo model in the filter degraded overall performance of the Kalman filter. He formulated several hypotheses to explain the finding, most of them based on the fact that the pogo model may be interacting with other states in the filter model. In his research to determine whether this was the case, he discovered that the atmospheric jitter model embedded in the Kalman filter was not performing as well as previously expected. In addition, his results indicated that there was probable interaction occurring between the pogo states and the jitter states that resulted in the degradation in performance of the 10-state filter compared to the 8-state filter. For this reason, he recommended that the plume pogo dynamics states be removed from the filter model until further studies could verify the interaction and a possible solution found.

During his thesis research, Rizzo also performed a stochastic observability test to gain insight into the relative observability of the states in his 10-state filter model [40]. He concluded that the filter contained a maximum of five states that were unobservable or only weakly observable in the output. Those states included the velocity and acceleration states characterizing the dynamics, and the pogo velocity state. This is a reasonable conclusion, since acceleration and velocity are not directly measured and would be more difficult to estimate in the chosen model. Because of the relatively benign trajectory being simulated for the ballistic missile, and the difficulty of estimating the acceleration states, Rizzo recommended that the target dynamics model be changed to a second-order shaping filter driven by white, Gaussian noise, along each axis (as opposed to the third-order model that included acceleration).

The recommendations made by Rizzo were implemented in the succeeding three theses [8,9,14], and the elemental FLIR filter that was used consisted of six states: four target dynamics states and two atmospheric jitter states. The velocity states were modeled as first-order lag filters driven by white, Gaussian noise. The simulations from the last three theses indicate that there is still a problem with estimating the atmospheric jitter states; specifically, the mean errors at (t_{i+1}^{-}) are much greater than the mean errors at (t_i^{+}). Therefore, the author was motivated to perform another series of stochastic observability tests on a series of filters with differing structures using the new target dynamics model (with no acceleration states) and comparing the results to tests with the original six-state target dynamics model. In addition, it was deemed useful to revisit filter structures that included the pogo states to gain further insight into how they may be affected (or not affected) by the modeling of the target dynamics and/or atmospheric jitter states.

For the MSOFE and AFIT elemental filter analyses, the target dynamics model was changed from $\dot{v} = -{}^{1}/_{\tau}\, v + w$ to $\dot{v} = 0 + w$, to allow for constant velocity paths better than those whose mean values would tend to decay toward zero over each propagation (as in the original six state model). Since this study was originally implemented to discover observability problems with the six state "FLIR filter" used by the previous three theses, that was the model studied. In the future, it would be useful to apply the same observability tests to the constant velocity models used later in this research.

5.2.2 The Stochastic Observability Test. If a stable, discrete-time dynamics/discrete-time measurement system of order n can be described by Equations (2-9)-(2-17), then the observability gramian matrix can be expressed as [21]:

$$\boldsymbol{M}_D(0,N) \triangleq \sum_{i=1}^{N} \Phi^T(i,0)\boldsymbol{H}^T(i)\boldsymbol{H}(i)\Phi(i,0) \qquad (5\text{-}1)$$

The discrete-time system is completely observable if and only if any of the following criteria are met: (1) the null space of the n x n matrix $\boldsymbol{M}_D$ is $0 \in R^n$ for some finite N, (2) $\boldsymbol{M}_D$ is nonsingular, i.e., invertible, (3) $\boldsymbol{M}_D$ is positive definite, (4) the determinant of $\boldsymbol{M}_D$ is nonzero [21]. If $\boldsymbol{M}_D$ is of rank $k<n$, then it is said that there are $(n - k)$ "unobservable states" in the model.

In order to gain some insight into the "relative observability" of the states, the magnitudes of the eigenvalues of $\boldsymbol{M}_D$ may be inspected. Theoretically, none of the eigenvalues should be negative, all should be real-valued, and there are as many unobservable states as there are zero eigenvalues. If any eigenvalue associated with a state is several orders of magnitude smaller than the other eigenvalues, that is an indication that the state is difficult to observe.

The observability gramian matrix and its eigenvalues were evaluated using the discrete Lyapunov equation solver in a controls design software packages called MATRIX_X [15]. The software was installed on a SPARC II workstation and utilized double precision numbers for all calculations. For independent verification of the results gained from MATRIX_X, the equivalent functions on an alternate software package called MatLab [46] were used. Both sets of tests yielded the same results. The next section presents the outcomes of the stochastic observability tests and the resulting conclusions.

5.2.3 Observability Test Results and Conclusions. The objectives in Chapter I outlined six different filter structures which were to be subjected to the stochastic observability condition. The structure of each filter is listed in Table 5.1. Each filter was to be analyzed with a known, constant target angle θ relative to the FLIR x-axis (set to 45 degrees or $\pi/4$ radians) and with a general angle derived from the velocity vector (as was done in the simulations). Recall the definition of the measurement matrix $\boldsymbol{H}_f$ (Equation 4-32) used in the measurement equations. In the constant velocity/constant angle case, the nonlinear elements defined in Equation (4-33) that

Table 5.1 Filter States for Stochastic Observability Tests

Filter	x_t/y_t	v_x/v_y	a_x/a_y	Atmosphere	x_p/v_p	Total States
A	2	2	2	6	-	12
B	2	2	2	2	-	8
C	2	2	-	6	-	10
D	2	2	-	2	-	6
E	2	2	-	2	2	8
F	2	2	2	2	2	10

are associated with the velocity states become zero. The nonlinear terms associated with the pogo state (H_{17} and H_{27}) become constants. Therefore, for the known constant angle cases, the linear Kalman filter update equations are used. For the general angle case, the extended Kalman filter update equations are used with the nonlinear measurement function $\boldsymbol{h}_f[\boldsymbol{x},t]$ and the $\boldsymbol{H}_f$ defined as in Equation (4-33) (i.e., the nonlinear terms are included).

For the case with the known constant angle θ, the stochastic observability tests yielded inconclusive results. The observability gramian matrix for all structures was found to be close to singular or badly scaled. Inspection of the elements of the matrices showed that the magnitudes varied by over 60 orders of magnitude with individual elements ranging from orders of 10^{-23} to 10^{44}. Of course, these eigenvalues that were many orders of magnitude smaller than the largest eigenvalues could be deemed essentially zero, indicating unobservable or essentially unobservable states. Moreover, some of the eigenvalues calculated were either negative or complex. These erroneous results are most likely the due to ill-conditioned Φ matrices, causing numerical precision difficulties with the software and hardware. It should be reiterated that all calculations were done in double precision (64-bit), which was the only avenue available to the author for avoiding such numerics problems. These first set of inconclusive results were obtained using the original sample period of Δt=1/30 seconds. The filters were also analyzed with the reduced sample period of 1/60 seconds to determine if there was any improvement in the scaling, but results were similarly inconclusive.

Since the numerical difficulties were thought to be a result of the large difference in orders of magnitudes of elements of Φ associated with the different states, it was a reasonable deduction that transforming the system with a scaling matrix before applying the observability test might yield better results. Several combinations of scaling matrices were applied and the observability gramian M_D and its eigenvalues calculated for each. Through use of the scaling matrix, the order of magnitude variation for the diagonal elements of M_D for some of the filters was reduced to about 20 (10^{20} variation). The associated eigenvalues for such cases were all positive and real, but showed similar large differences in orders of magnitude. It was hoped that groupings of eigenvalues, some large and others many orders of magnitude smaller, would provide some insight into unobservable or weakly observable states, since the eigenvalues in the smallest groups can be declared essentially zero. However, as emphasized before, even the groupings resulting from use of the scaling matrix varied so extensively in magnitude (beyond the precision of the machine implementing the test), that it would probably be overly speculative to make any definite conclusions. Therefore, these results were also considered inconclusive.

Because the observability issues were unresolvable for the known, constant θ angle case, no attempt was made to subject the filters to the observability conditions for the general angle-of-attack case. Because of the insurmountable numerical precision problems, the only other way to gain insight into the observability of the states is to analyze the statistics of Monte Carlo simulations. If better hardware and/or software are made available to researchers in the future, it may be beneficial to attempt the stochastic observability studies again.

5.3 MSOFE Monte Carlo Analysis

The Multimode Simulation for Optimal Filter Evaluation (MSOFE) program [6] is a package coded in the FORTRAN software language, that can perform both covariance and Monte Carlo analyses on single linear or extended Kalman filters. The code is self-documented and constructed in a modular form so that the truth system and filter models are programmed in subroutines accessible to the user. The core program that performs the analysis is in a separate package and, for this research, was not altered. The initial conditions and tuning values for the truth and system filters are contained in a data input file, enabling runs to be performed without re-compiling and linking the code. MSOFE produces data listings which are processed through another program called MPLOT, which produces data files compatible with numerous different plotting programs. The plots appearing in this thesis were produced using MATRIX$_x$ [15].

A considerable amount of time was spent becoming familiar with MSOFE and creating and debugging the continuous-time truth and filter models in the code. Now that the basic models have been established, continued research on with this package should be somewhat simplified since only slight modifications to the analysis process need be made. This section presents the procedures used for analyzing the filters using MSOFE and the results obtained from that analysis.

Since any explicit conclusions drawn from the stochastic observability studies would be speculative, it was anticipated that analysis of some of the same filter structures through Monte Carlo simulation may prove more fruitful. The primary objective of the analysis was two-fold: first to investigate the estimatability of the atmospheric jitter and plume pogo; and to determine if there were any degrading affects caused by interaction of the atmospheric jitter and pogo states. Another reason was to check out the filter performance independently of the "AFIT software" to ensure that performance attributes (e.g., poor jitter estimation at t_{i+1}^- vs. good estimation at t_i^+) were really inherent *in the problem* and *not* due to some quirk in the simulation (especially of the truth model) in the "AFIT software". For these reasons, the emphasis of the analysis presented in this section has been placed in those areas. Also, because of the unforeseen time expended in the familiarization and debugging process, the number of filter configurations originally established for analysis in the objectives (see Table 1.2) was reduced to the ones which the author felt would yield the most insight into the estimatability of jitter and plume pogo and assist in analyzing the performance of the elemental filter. This section describes the results of the 10-run Monte Carlo analysis performed using MSOFE. Appendix A contains a brief discussion of how the statistics in this chapter and the next were calculated.

5.3.1 MSOFE Filter Configurations. Table 5.2 shows the subset of the system/filter combinations enumerated in Table 1.2, which were subjected to Monte Carlo analysis with MSOFE. Note that, as mentioned in the chapter's introduction, the LEL measurement is not incorporated into any of the models and therefore the offset state is not included. Engineering intuition and past experience (especially Rizzo's results [40]) lead one to anticipate observability problems without the LEL measurements. It is anticipated that inclusion of the LEL measurements should assist the filter in estimating the more weakly observable states.

Table 5.2 Filter/Truth Model Combinations Analyzed with MSOFE

	TRUTH MODEL					FILTER MODEL			
SIM	N	DYN	POGO	ACCEL	JITTER	JITTER	DYN	POGO	N
2[1]	6	4			2		4		4
4	6	4			2	2	4		6
6[2]	6	4	2				4	2	6
7[1]	8	4	2		2		4		4
8[1,2]	8	4	2		2		4	2	6
9[1]	8	4	2		2	2	4		6
10[2]	8	4	2		2	2	4	2	8

Notes:
1. Filter structure does not match truth model.
2. Analysis performed for both known constant and general angle cases.

Combinations 1, 3 and 5 from Table 1.2 were not analyzed because the filter contained states (pogo) that were not included in the truth system. The last six combinations from Table 1.2 were not analyzed because they contained acceleration states in the truth model. Since the intention was to use the MSOFE analysis as a stepping stone to analysis of the elemental filter, which did not contain acceleration states modeled in the dynamics, those runs were considered to be of low priority. Table 5.2 shows that 4 different filters were tested against three different truth models. All seven combinations in the table were tested with a deterministic constant angle between the velocity vector and the positive FLIR x-axis ($\theta_f = 60°$). The simulations 6, 8 and 10 in Table 5.2 were also analyzed with the general angle model in the filter. (Recall that the difference is in the update model, which becomes linear for a constant known angle θ_f, but nonlinear for a general angle, defined in terms of the filter estimated velocities v_x and v_y, which breaks the pogo into the FLIR axes x and y components; recall section 4.3.1.3.)

Six different pogo conditions were also considered for study to determine whether the magnitude or frequency of the pogo oscillation affected estimatability of the pogo and/or atmospheric jitter. The conditions are listed in Table 5.3. Since simulation number 10 of Table 5.2 most closely approximated the simulation in the AFIT software, only that filter was analyzed under all six pogo conditions. Pogo condition four, considered nominal for MSOFE analysis, was simulated in the truth and filter models for the filters analyzed with the known constant angle θ_f (as denoted by Note 2 in Table 5.2).

In the MSOFE Monte Carlo simulations, as with the elemental filter evaluations of the next chapter, the simulation coordinates are in terms of the FLIR plane. The filter states were initialized to the true conditions in order to study tracking performance (as opposed to acquisition

Table 5.3 Pogo Conditions for Nonlinear Filter Analysis

Pogo Condition	Frequency (Hz)	σ_p^2 (pixels^2)	Gain Constant K_{pf}	Initial Pogo Covariance
1	0.1	0.067	0.03780	4.489e-3
2	0.1	0.670	0.37800	4.489e-1
3	1.0	0.067	0.01195	4.489e-3
4	1.0	0.670	0.11950	4.489e-1
5	10	0.067	0.00378	4.489e-3
6	10	0.670	0.03780	4.489e-1

performance from poor initial conditions). Based on a nominal velocity of 2700 meters/second over the 10-second Monte Carlo simulation at an initial velocity angle of 60°, the component velocities were initialized to v_x = 450 pixels/second and v_y = -779.4 pixels/second. All other initial state conditions were established at zero. The variance for the atmospheric jitter process in both FLIR plane directions was $\sigma_a^2 = 0.2$ pixels^2, the same as in the elemental filter. Unlike the elemental filter simulation, the measurement noise covariance is considered equal in both axes (as might be expected in a more modern FLIR pixel array). The measurement noise variance $\mathbf{R}_f$ (with units of pixels^2) associated with sensor noises was:

$$\mathbf{R}_f = \begin{bmatrix} 0.00436 & 0 \\ 0 & 0.00436 \end{bmatrix} \tag{5-2}$$

Nominal range was set at 200 Km and the same pixel proportionality constant k_p = 15 μrad/pixel was used. The choice of a decreased value for the nominal range was to reflect a scenario similar to the actual scenarios anticipated by the sponsor at the Phillips Laboratory [4,5]. The intention was eventually to modify the simulations in the AFIT software to reflect similar scenarios.

5.3.2 Monte Carlo Analysis Results. Since sensitivity analysis using MSOFE was not intended to be exhaustive, but only to provide insights into obvious filter performance deficiencies and state estimation interrelationships, this section will not list exact statistics for individual runs. Also, to avoid redundancy of detailed observations (many simulations yielded nearly identical results), it will discuss the general observations resulting from inspection of the plots obtained from the simulations.

Sample plots that show specific features emphasized in the discussion are contained in Appendix B. Figure B.1 is a typical sample of the plots produced from a simulation. The top plot, Figure B.1(a), shows a direct comparison of the ensemble average truth state and ensemble average filter estimated state (averaged over 10 runs). The solid line represents the truth and the line of evenly spaced dashes represents the filter estimate. The bottom plot, Figure B.1(b), shows a comparison of the statistics calculated for the system and the filter over 10 runs. The solid line denotes the mean error (ensemble average of filter estimate minus true value). The evenly spaced dash lines are the mean filter-computed state error standard deviation. The dotted lines denote the mean plus and minus one standard deviation calculated from actual errors. Good tuning is indicated by $0 \pm \sigma_f$ enveloping the mean $\pm\ \sigma$ of the actual errors calculated from the Monte Carlo analysis. The plots for the simulations discussed in Section 5.3.2 are contained in Appendices C and D.

5.3.2.1 Linear Filter (Constant θ_f) vs Nonlinear Filter. As previously mentioned, configuration numbers 6, 8 and 10 in Table 5.2 were analyzed with nominal pogo conditions for both a general and constant angle. Inspection of the plots for all three simulations showed virtually no differences between the general angle and constant angle. These identical results verify the correctness of the extended Kalman filter update cycle used for the nonlinear, general angle case. The identical results also indicate that the three nonlinear filter configurations estimate the velocity angle θ_f very well. Note that such accurate angle estimation is *not necessarily* an indication of how well the individual velocity states are estimated, but is a reflection of the *ratio* of the estimated velocity states, since the angle θ_f is a function of that ratio. With the exception of the analysis dealing with different pogo conditions, all of the following results and conclusions were drawn from comparison of the linear Kalman filters.

5.3.2.2 Position Accuracy. The most accurate filter estimation occurred when both the truth and filter system did not contain atmospheric jitter states and matched each other in structure. Simulation six demonstrated the highest position estimation accuracy (0 mean errors, $1\sigma \approx 0.04$ pixels). The filter was well tuned in all states. These results were expected and serve as a good baseline for filter performance against a real world without jitter. This is not realistic, but leads to the conclusion that, because estimation of pogo was fairly accurate (zero mean, $\sigma \approx 0.04$ pixels), the pogo model is sound and the pogo states are estimatable.

The most accurate filters when atmospheric jitter was simulated in the real world, were the filter models in simulations 4 and 10. In these simulations, the filters are well tuned in all states and deliver equivalent accuracies with zero mean errors and $\sigma \approx 0.18$ pixels. In both simulations, the high accuracy would be expected since the filter structure matches the truth model. However, simulation 10 includes pogo states and simulation 4 does not. Analysis of simulation nine also indicates equivalent position accuracy, despite the fact that the truth model simulates plume pogo but the filter does not estimate it. Apparently, the errors due to the pogo phenomenon are attributed to atmospheric jitter and appropriately compensated by the filter.

5.3.2.3 Atmospheric Jitter Estimation. Comparison of the simulation data shows that atmospheric jitter is the dominant effect on filter accuracy. All filter structures that didn't include jitter when the truth model did (simulations 2, 7 and 8), drastically underestimated their own errors and demonstrated poor accuracy relative to filters that estimated jitter. Errors were zero mean, but standard deviations were twice as large ($\sigma \approx 0.3$ to 0.4 pixels). Since total real jitter has $\sigma_a^2 = 0.2$, or $\sigma_a \approx 0.45$, standard deviations of such a magnitude indicate large estimation errors. In all three simulations, the filter attempted to compensate for unmodeled jitter errors by overestimating changes in velocity (all three cases) and overestimating pogo oscillation magnitude (simulation 8). Again, inclusion of pogo in either the truth model, the filter model, or both did not noticeably affect overall position errors. No effort was made to tune the filters since that was not the purpose of this exercise. With tuning, one could get more correct indications of trends, but *not* including jitter states in the filter is *not* a good idea.

When properly tuned, the filter was able to estimate jitter existing in the truth model with zero mean errors and $\sigma \approx .25$ (see simulations 4, 9, or 10). The plots of the jitter statistics show that filter-computed error estimates at $(t_{i+1}^-)(\sigma \approx 0.30)$ are still greater than filter-computed error standard deviations at $(t_i^+)(\sigma \approx 0.20)$ by a factor of about 0.5. This is similar to the estimation problem found by Rizzo [40], but not nearly to the same extent. Rizzo observed factors as high as several orders of magnitude [24]. This discrepancy between the MSOFE and AFIT software should be resolved. The filter does benefit by including the jitter states, since the uncertainty in position caused by the jitter process itself is 0.4472 pixels. By including the jitter states, the filter can decrease the uncertainty in position by about one half.

5.3.2.4 Plume Pogo Estimation. All of the MSOFE simulations indicated that the pogo phenomenon is not inherently difficult to estimate. Simulation 6 showed that if it is the only disturbance on the centroid position (other than target dynamics), it is easy to estimate. Simulation 9 showed that the filter is robust enough to compensate for errors due to pogo by attributing them to jitter. However, simulation 8 showed that the converse is not true; inclusion of pogo states cannot compensate for the existence of atmospheric jitter. Simulation 7 showed that performance is very poor if neither jitter nor pogo are included in the filter model.

Inspection of the pogo state plots from simulation eight show that the filter attempts to compensate for errors due to lack of jitter estimation in its structure by increasing the pogo oscillation magnitude. Since the power spectra of pogo and atmospheric jitter processes are different (specifically, jitter has a higher break frequency than pogo and can appear almost white in comparison to pogo, which shows a resonant power spectral density peak around its undamped natural frequency) all that increasing the estimated magnitude serves to accomplish is to degrade pogo estimation. Resulting position accuracy is not improved. It was felt that tuning the filter (varying the dynamics noise and/or measurement noise used for the filter model) may assist in improving the pogo estimation accuracy. A scaling factor was applied to the pogo process noise strength in the filter (which was established to the same value as the truth model) to determine whether better tuning would reduce the mean standard deviation of actual errors. It was found that the best scaling factor to use was unity (i.e. tell the filter what true conditions are).

5.3.2.5 Plume Pogo/Atmospheric Jitter Interaction. The preceding analysis shows that pogo is less difficult to estimate than jitter. In fact, the presence of the jitter model seems to be the dominating factor in how accurately the filter estimates position. If the filter model includes jitter estimation, the limiting factor on position accuracy is directly dependant on how well jitter is estimated. Analysis does not indicate that there is any major degrading type of interaction between the pogo and jitter models. Atmospheric jitter estimation accuracy did not degrade when the pogo model was introduced (simulations 4 and 9 vs simulation 10). On the other hand, pogo estimation improved when the atmospheric jitter states were added to the filter (simulations 6 and 8 vs simulation 10).

5.3.2.6 Sensitivity to Different Plume Pogo Conditions. This subsection presents observations based on analysis of the nonlinear eight-state filter used in simulation 10 from Table 5.2. A ten-run Monte Carlo analysis was performed on the filter for each of the six pogo conditions listed in Table 5.3. In each case, the filter was tuned for the proper pogo conditions. Inspection of the plots in Appendix D shows that all the mean errors in the position states were approximately zero. The runs for pogo conditions 1 and 3 show that the x position mean error may be slightly negative (on the order of -0.04 pixels), but this could be a reflection of an insufficient number of Monte Carlo runs. Under all pogo conditions except number 2, the x and y position errors were the same: zero mean, filter-computed standard deviation $1\sigma \approx 0.19$ pixels, actual standard deviation $1\sigma \approx 0.19$ pixels. In other words, the filters appear to be well tuned. Pogo condition 2 is the exception. Position errors were zero mean and the filter was well tuned, but 1σ for x-axis errors was about 0.34 pixels and 1σ for y-axis errors was about 0.53 pixels. This implies that slow oscillations of large magnitude are the most difficult to estimate.

As with position accuracy, estimation of atmospheric jitter was the same for all pogo conditions. Atmospheric jitter mean estimation error was zero, with filter-computed error and actual standard deviations of about 0.25 pixels (the same as for the best estimation of the linear filters).

Note that the upper plots (figures (a)) in Appendix D are the mean over 10 runs as opposed to mean ± 1σ. If the truth model initial conditions had been randomized, the mean could be zero for all time, while a steady-state error standard deviation (not shown in the plots) in consonance with the pogo process σ_p^2 would be achieved. Initial condition randomization should be accomplished for more realistic simulations. It is suspected that the filter may be fairly well tuned but randomization of the initial conditions would allow a better performance assessment. Also note that there is an *apparent* anomaly on the pogo offset plots for pogo conditions three through six -- truth model oscillations die out vs maintaining the σ_p^2 of Table 5.3. This actually is *not* an anomaly and should be expected, since the mean pogo value *should* appropriately die out (as mentioned earlier in this paragraph). The plots appear with a nonzero mean because the initial conditions for magnitude and phase were established to identical values at for every run.

Table 5.4 Pogo Error Statistics for Different Pogo Conditions

Pogo Condition	Mean Error	Filter Computed σ	Actual σ
1	0	0.10	≈ 0.100
2	0	0.50	≈ 0.500
3	0	0.01	≤ 0.005
4	0	0.10	≈ 0.100
5	0	very small	very small
6	0	very small	very small

As previously mentioned, if the initial conditions were randomized, the pogo process would have the appropriate zero mean and standard deviation from the start of the run. The upper plots for pogo condition should not be misunderstood, since they could be misleading without the randomized initial condition and the ± $1\sigma_p$ traces.

Plume pogo estimation statistics varied, depending on what condition was in existence. Table 5.4 shows the approximate statistics for the pogo states of the filter tuned for differing pogo conditions, obtained off the plots in Appendix D. Note that the errors become negligible when the pogo oscillations are at the highest frequency of 10 Hz.

5.4 Summary

This chapter presented the procedures, filter models and results of the stochastic observability tests. The tests proved to be inconclusive due to numerical precision difficulties. Section 5.3 explained what filter configurations were subjected to Monte Carlo analysis with MSOFE and the results of the performance analysis. The accompanying plots for the linear Kalman filter discussion are contained in Appendix C. The plots that accompany the description of results for the nonlinear filter tuned for alternate pogo conditions are contained in Appendix D. An exhaustive run by run description of the plots is not contained in this section, but the plots are included so that the interested reader may scrutinize the plots further in order to draw their own conclusions.

VI. AFIT Elemental Filter Analysis Procedures and Results

6.1 Introduction

The software simulation of the real world, measurement generation, enhanced correlator algorithm and various Kalman filters have been developed at AFIT over the last 14 years [31]. The truth models used to simulate the real world, and the generation of the measurements for the tracking filter, are described in Chapter III. A detailed description of the nine-state dynamics model and the measurement model of the filter which is analyzed in this chapter is contained in Chapter IV.

This chapter presents the process used for analyzing the elemental filter in the AFIT software. Recall from the objectives in Chapter 1 that the fundamental goal of this portion of the research was to combine the two filters (one for tracking the center-of-intensity of the target/plume IR image, and one for estimating the offset between that center-of-intensity and the center-of-mass of the missile hardbody) developed and analyzed by the previous three researchers, Eden, Evans and Herrera [8,9,14], into a single filter that also incorporated estimation of the plume pogo. The intention of this goal is eventually to determine tuning constants for a series of filters, each of which is tuned for optimal performance when exposed to different pogo characteristics (in this instance, rms amplitude and frequency of pogo oscillation). That series of filters would then be used to form a multiple model adaptive estimator for this application.

The following sections describe the performance delivered by an elemental filter tested against nominal pogo conditions. The associated performance plots, displayed in Appendix E, reveal that there may be errors in the software causing inaccurate simulations under certain conditions. These potential errors, their effect on the simulation, and proposed solutions are also described in the ensuing sections. Although numerous runs were made under many different conditions, a single run was selected for Appendix E that contains all the necessary salient features found throughout all the runs that will be addressed in the detailed analysis.

The elemental filter which produced the results described below contains nine states and updates the state vector from two separate sensors. The first update contains plume centroid position information from the FLIR. Raw FLIR data is processed through the enhanced correlation algorithm and position of the plume centroid is presented to the filter in terms of two offsets from the center of the field-of-view corresponding to FLIR plane x- and y-axis coordinates. After the filter performs an update to the state vector, a low-energy laser sweep is initiated in order to obtain a measurement of the center-of-mass location relative to the plume centroid. The sweep is oriented to start at the estimated centroid location, calculated from the filter's state vector, and to intercept the hardbody along the velocity vector, also calculated from the state vector. The Doppler frequency shift characteristics in the LEL return are exploited to determine the offset distance of the hardbody center-of-mass from the plume center-of-intensity along the velocity vector. The Kalman filter then uses this measurement to perform a second update before estimating the position of the plume centroid and center-of-mass at the next measurement period. The FLIR plane is shifted in order to center the FOV on the centroid estimate at that next sample period.

6.2 *Performance Analysis Results*

Appendix E contains selected plots which depict the performance of the elemental filter for a single simulation of five Monte Carlo runs with nominal pogo conditions. The filter pogo gain constant, K_{pf}, is adjusted to reflect the desired pogo rms amplitude of 0.112 pixels (i.e., the filter dynamics noise strength is set equal to the truth model's). The other tuning parameters and simulation conditions are listed in Chapters III and IV. Table 6.1 lists the performance statistics associated with the plots. Two different types of plots are contained in the Appendix E. The first type directly compares the value of the filter estimated state to the true state. As in the previous appendices, the solid line is the true condition and the dashed line is the filter estimate. The other type of plot contains five traces; these are the statistics plots explained in Appendix B.

Table 6.1 Time-Averaged Error Statistics for Elemental Filter (Nominal Pogo Conditions)

STATE	MEAN ERROR(t_i^-)	MEAN ERROR(t_i^+)	STD. DEV. $\sigma(t_i^-)$	STD. DEV. $\sigma(t_i^+)$
X POSITION	-0.489990	-0.488290	0.43488	0.43263
Y POSITION	0.748290	0.745320	0.41595	0.41270
X VELOCITY	-0.194090	-0.188110	0.53733	0.53703
Y VELOCITY	0.287940	0.276900	0.53360	0.53473
X JITTER	0.024115	0.034851	0.40936	0.35582
Y JITTER	-0.047987	-0.071015	0.45138	0.42079
POGO OFFSET	-0.007040	0.006787	0.13381	0.13382
POGO VELOCITY	-0.049104	-0.047486	0.81970	0.81876
OFFSET	-0.744800	-0.750030	0.92680	0.93146
X CENTROID	-0.110370	-0.106100	0.52676	0.52670
Y CENTROID	0.099324	0.092069	0.82138	0.82417

6.2.1 Position Estimation Accuracy. Estimation accuracy of the position is the primary measure of overall filter performance. There are two important features that should be noticed in the x and y position error statistics plots (Figures E.1 (a) and (b)). The first is that the filter is not properly tuned for the position states, despite that fact that it was "told" the proper pogo conditions to expect and also the fact that the optimal tuning parameters for atmospheric jitter, determined in previous theses [17,38,47,40], were used. Since estimation accuracy of the other states will affect the position accuracy, inspection of the other statistics plots shows that the poor tuning is partially impacted by sub-optimal tuning in some of the other states. The other important salient feature is that there is a nonzero mean error in both the x and y directions. Possible causes of the biased errors will be presented later in the analysis.

6.2.2 Velocity Estimation Accuracy. An important feature present in the velocity error statistics plots (Figure E.2) and all the other statistics plots (except for the plume pogo statistics) is the transient during the first second of the simulation. The actual errors were off the scale of the plots shown, so statistics were not plotted until after the beginning of the simulation. The actual filter velocity estimate was initialized to the true velocities, so errors are zero at t_0. A frame by frame inspection of the velocity data showed that an unknown phenomenon is causing filter velocity estimates to suffer an extreme transient in the first two to five frames before recovering. The fact that the filter is able to recover from such a large transient in the velocity state is an indication of robustness, despite its constant velocity model. The unknown anomaly causing the initial velocity transient is also the probable cause of the initial transients visible in the other states. The error statistics in Table 6.1 were accumulated from two seconds into the simulation to the end, so they should be an accurate representation of the filter's steady state performance.

Other than the initial transient, velocity tracking is reasonably good and filter estimation tuning is close to optimal; a slightly higher noise strength may be beneficial to optimize the tuning. Mean errors appear to be close to zero and exhibit no obvious systematic nonzero trends that would require attention through tuning or modeling effects.

6.2.3 Atmospheric Jitter Estimation Accuracy. The next two figures (Figures E.3 and E.4) show a visual presentation of the filter's estimate of plume intensity centroid displacements in both axes due to atmospheric disturbances. Performance was approximately equal along both axes. The errors don't appear to be associated in any way with the pogo state plots, so there is no *apparent* affect of pogo coupling into the atmospheric jitter states. As noted in the previous theses [8,9,14,40], there is a distinct difference in filter estimated uncertainty immediately before and after the update period, however, the difference in this case is on the order of only 50 percent of the 1σ estimate at t_{i+1}^{-}. Previous observations showed a difference of several orders of magnitude [24]. The time-averaged filter 1σ is about 0.25 pixels, but *actual* errors exhibited temporally averaged standard deviations on the order of 1σ = 0.4 pixels in each direction (see Table 6.1). These errors indicate that the filter is not estimating the jitter very well at all, since the actual jitter process itself had a variance of 0.2 pixels^2 (or σ = 0.447). The filter could possibly track as well without any jitter model states at all. This observation is contrary to the results obtained from the MSOFE analysis, in which estimation of jitter was not only possible, but significantly improved filter performance when included in the model.

6.2.4 Plume Pogo Position and Velocity Estimation Accuracy. The two figures showing the filter's plume pogo offset position and velocity estimation performance (Figures E.5 and E.6) indicate there is room for significant improvement. The bottom plot on E.5 and E.6 indicate a *nonrandomized initial condition* on truth model simulations of pogo (so *all* samples show the same phase, giving the sinusoidal mean motion that is obvious on the plots). This sinusoidal mean would be removed if the initial conditions were randomized. It is then suspected that the performance plots would indicate reasonably good tuning and performance. There is apparently a nonzero initial offset in the filter estimate (Figure E.5(a)), however, the pogo offset should be showing zero, as it was set in the software. A frame-by-frame analysis would be necessary to determine the actual initial setting. There could be a large initial transient similar to the observation associated with the velocity states. The initial estimates of the filter cause the pogo position (and velocity) estimates to be out of phase with the truth model, and hence the major contributor to pogo estimation errors. By the end of the 6.5 second simulation, the filter is beginning to recover and reach steady state. The general trend of the plots shows that the filter would probably be well tuned in a steady state condition. It is also probable that it would recover more quickly if the pogo oscillation frequency were higher. This hypothesis is supported by the MSOFE data shown in Section 5.3.2.6 that indicates that higher frequency and lower magnitude pogo conditions are easier to estimate. The corresponding plots of pogo estimation from the MSOFE analysis also showed that estimation can be very accurate if the filter is in the proper phase. The initial conditions of no pogo offset and zero velocity do not reflect conditions that would ever exist in the real world and would be cause for concern about model adequacy if they were coupled with small initial variances. However, use of large initial filter covariance terms P_o for pogo position and velocity would be imitative of the real world (symbolizing high uncertainty of initial conditions), and *mean* values of zero for both states would be fine initially.

This simulation, however, is important in that one cannot expect the filter to receive the proper initial pogo conditions and the filter should be able to recover from improper conditions.

It is useful to note that recovery time for this model for these pogo characteristics is fairly extensive (over 6 seconds). This simulation also showed (as did the MSOFE simulations) that the filter recovers from magnitude errors more rapidly than from phase errors. This suggests that large pogo P_o and Q values could be used for an initial acquisition phase to speed up the transients. Then, the Q values used for this simulation could be used in track mode. Finally, the pogo errors do not appear to be modulated in any way by the atmospheric jitter estimation errors, indicating that there is probably no corruption from the jitter states.

6.2.5 Offset Estimation Accuracy. Figures E.7(a) and E.7(b) show the filter's estimate and error statistics respectively for the offset distance between the plume intensity centroid equilibrium point and the missile center-of-mass. The plots clearly show that tuning of the filter leaves much to be desired. Also note that there is a substantial bias in the mean error and that the filter's estimate is not constant, as it should be. Since the offset is modeled as a constant in the dynamics model, the high frequency variations in the offset value are probably due to the state updates from the low-energy laser measurement. It is suspected that much of the degradation in filter performance can be attributed to the offset measurement. The offset error contributes directly to the bias in mean error previously highlighted for the position states (Section 6.2.1). The justification for this conclusion is revealed by analysis of the centroid estimation accuracy. The erroneous offset measurements are described and several possible explanations for the poor offset performance are proposed in Section 6.3.

6.2.6 Plume Intensity Centroid Estimation Accuracy. Although position of the plume intensity centroid is not explicitly modeled in individual states within the filter structure, it is easily calculated from the state vector and is carried throughout the software simulation. In fact, until this thesis, the filter position states defined the centroid location, since all enhanced correlator, template generation, and FLIR controller functions are based on the centroid's position. Because of the importance of the centroid location estimates, an analysis of the filter's estimation accuracy of them has been included here. The last figure in Appendix E shows that tracking of the plume centroid has essentially zero mean. Since the current filter defines the target position states as the missile center-of-mass, and they are calculated relative to the centroid position, the bias in the position states (mentioned in Section 6.2.1) must be caused by the biased offset measurements. There are no traces indicating filter-computed error uncertainty since the centroid locations are not explicitly represented in the filter state vector. Filter computed covariance could be calculated through use of the transformations CP_fC^T, where $y = Cx_f = x$ and y centroid position coordinates, however for this thesis, the ultimate measure of performance is determined by the accuracy of the center-of-mass location. If centroid location tuning becomes an issue, it may be advantageous to incorporate the transformation equations into the software.

6.3 Potential Problems and Proposed Solutions

This part of the chapter is divided into two subsections. The first presents explanations of problems detected during the performance analysis of Section 6.2 The second section addresses several miscellaneous issues that the author felt impelled to record for the benefit of future researchers.

6.3.1 Specific Problems Revealed by Analysis. One can find reason to suspect many errors in the simulation from analysis of the center-of-mass to centroid equilibrium point offset plots. Recall that the measurement from the LEL was considered highly accurate when present (a low measurement noise variance R relative to the dynamics noise strength Q). A frame-by-frame analysis revealed that offset values from the laser measurement did indeed appear to vary more rapidly than expected (even with pogo taken into account) and that there were possible errors in the laser sweep routine. Both the bias and the evidence of noisy laser measurements observed in Section 6.2.5 are reason to suspect that there are errors in the measurement generation routines in the AFIT software.

The poor laser measurements lead one to speculate that a combination of events may be occurring in the software. First, the bias may be a reflection of an improperly modeled Doppler measurement. The Doppler measurement model used was developed and shown to generate unbiased measurements successfully by Herrera in the previous thesis [14]. However, it may have been implemented improperly in this research due to errors when transferring software code, improper tuning (recall from Chapter III that the Doppler measurement is dependent on several factors) or any other misunderstanding of its implementation. The existence of the bias, which is of the same approximate magnitude and direction characteristic of the speckle return model [14], indicates that the Doppler measurement model should be inspected again to ensure proper implementation.

Another event that may be causing both the bias and the noise is the laser sweep/scanning routine. The frame-by-frame analysis showed that there were extended periods up to one second during which no updates were made from the low-energy laser due to a lack of intercepting the hardbody. This high miss-rate was not in consonance with the 0.1 percent probability of miss and benign Doppler return parameters used for the simulation. Also, during the course of running a few other simulations, particularly those with larger pogo magnitudes, the simulation locked itself into an endless loop inside of the sweep routine. There are at least two potential causes for this error: LEL sweep geometry, and FOV geometry. The first cause, laser sweep geometry, is highly likely and was originally proposed by Evans [9]. It was the reason the LEL measurement routine was upgraded from a simple single, scan routine to a series of scans forming the sweep. The filter's estimate of the centroid location is not exact, and therefore basing the laser sweep on that information may still cause either a partial (only part of the hardbody is intercepted by the sweep) or complete miss. When Evans developed the sweep routine, he optimized it for a high LEL/Hardbody intercept rate without any plume pogo. It may be necessary to modify the sweep geometry (perhaps a simple expansion of the sweep angle) to improve the chances of intercepting the entire missile hardbody in the presence of plume pogo. An excellent detailed description of the scan and sweep routines is contained in Eden's and Evans' theses [8,9].

In addition to erroneous measurements caused by sweep geometry, the poor intercept rate may have been caused by the geometry of the FLIR tracking window itself (the FOV). The increased frequency of problems when higher pogo magnitudes were tested leads to speculation that part of the missile hardbody may actually be projected outside the borders of the FOV. Since the FLIR controller seeks to center the FOV on the centroid, the FOV will oscillate along with the plume, so the projection of the missile hardbody is what actually oscillates in the FLIR plane. The plume pogo distance, centroid equilibrium point offset distance, and missile length may combine during the simulation so that part of the missile hardbody projection exceeds beyond the boundaries of the FLIR plane. Further frame-by-frame analysis and creation of more statistics would be required to verify if this is occurring.

One ad hoc software patch that decreases difficulties caused by geometry problems (and which was implemented in some simulations for this research) is simply to declare that the sweep did not intercept the hardbody. This patch prevents the simulation from locking into an endless loop, but does not prevent a low intercept rate or fix either the laser sweep or FOV geometry problem. A potential fix to the FOV geometry problem, if it is verified as being a cause of difficulties, would be to utilize a larger FOV. This could be done by expanding the dimensions of the tracking window to more than 8 x 8 pixels. Another method that expands the effective tracking window is diagonally rotating the FOV (DRFOV) so that the velocity vector is aligned at 45° to the FLIR plan axes. This rotation effectively expands the length of the combined plume and hardbody that may be projected in the FOV and increases the chance of maintaining the track. This method was successfully implemented and tested (with good results) by Rizzo [40], but prior to the implementation of the LEL measurements and projection of the hardbody on the FLIR plane. It may be advantageous to investigate using the DRFOV routines with the new elemental filter developed in this thesis.

6.3.2 Miscellaneous Issues. In the course of this research, an inordinate amount of time was spent becoming familiarized with the software. Because of its rapid development and numerous contributors, much of the internal documentation (the only current documentation available) has become outdated. This is complicated by the fact that the software has been developed over a long length of time and has an extremely high number of possible variations. Consequently, it has become a highly complex and admirable accomplishment, yet one that can still be improved tremendously. The inherent modularity of the structure should be maintained and perhaps even increased. By utilizing function modules and subroutines, the variations in configuration are easier to keep track of and to implement. Some of the features of MSOFE, such as common variables and user designated parameters, were incorporated into the software and have made debugging and future changes to the model much simpler and efficient.

Because of its rapid development, many errors were found in the software that did not cause problems in previous theses, but were brought to the surface during this development. Chief among these were some sign errors arising from the use of a right-handed convention for aligning the FLIR plane-to-missile range vector. Although this would initially be a logical choice of coordinate systems, it causes a discrepancy between the orientation of the truth and filter FLIR plane coordinates along the *y*-axis (see Chapter III). Furthermore, once the initial projection of the missile and plume are made onto the FLIR plane, all calculations and control functions are performed in terms of that coordinate system. This necessitates extraordinary care and attention to the signs of any calculations involving *y*-axis components. Use of a left-handed coordinate system, so that the range is considered negative, would allow the truth and filter coordinate systems to coincide with each other. This would greatly simplify the debugging process for sign errors and make the code more efficient.

Since the currently implemented truth model was originally developed when only FLIR measurements were available, the target dynamics states are defined in terms of the plume intensity centroid. The offset between the intensity centroid and the hardbody center-of-mass is hardcoded as a constant (not explicitly modeled) and the center-of-mass location is a derived value calculated from the truth state vector. This definition is convenient when using a structure with two independent filters (as in the previous thesis), but is inefficient for the development of this thesis and also does not reflect reality. With the incorporation of LEL measurements, information is now available to locate the true target of interest, the missile hardbody. For this reason, the truth model should be revised so that the dynamics states describe the missile center-of-mass. This would also necessitate modeling the offset in the truth model, as opposed to hardcoding it to a constant. The revision should yield a more realistic and logical truth model which leaves more room for future embellishment. For example, the offset could be modeled as a variable or unknown distance. Revision of the truth model will also bring it more into line with the structure anticipated for future elemental filters.

Another residual of the original FLIR measurement-only structure of the truth model is that the FOV, defined in terms of the FLIR plane, is centered on the plume intensity centroid. As mentioned previously, this causes the FOV to oscillate with respect to the missile hardbody (or vice versa, depending on how you look at it). Although controller delay times and servo limitations were regarded as negligible in earlier theses (see Section 2.3.3), they may once again become complicating factors that resurface with the new measurement, revised truth model, and higher sample rate. It is probable that a much smoother signal would be delivered to the pointing controller if it is directed relative to the hardbody instead of the centroid. By the same token, it would be a logical step to revise the FOV to center on the estimated center-of-mass. This revision would require detailed investigation of the enhanced correlation algorithm, template generation, and template re-acquisition routines to ensure that they continue to operate as designed.

6.4 Summary

This chapter presented the results obtained from analysis of the AFIT elemental filter simulation. A single simulation was selected and a detailed explanation of filter accuracy given for each state. The figures shown in Appendix E indicate that possible errors still exist with the currently implemented design. Explanations of those possible problems and potential solutions were offered in Section 6.3, as well as a discussion of several miscellaneous issues surrounding the AFIT simulation software.

VII. Conclusions and Recommendations

7.1 Introduction

This chapter summarizes the final conclusions of this thesis and suggests areas for further study. Section 7.2 draws conclusions based on the results obtained in Chapters V and VI. Section 7.3 enumerates topics that require further research either to resolve problems found in this thesis or to further the development of the general ballistic missile tracking objective.

7.2 Conclusions

Numerous conclusions were drawn, based on the results of the observability tests, MSOFE analysis, and elemental filter analysis. The conclusions contained in this section are summarized, so the reader should refer to the specific sections in those Chapters for detailed discussions.

7.2.1 Stochastic Observability Tests. The results of the stochastic observability tests explained in Section 5.2.3 were inconclusive. Six different filter structures consisting of different combinations of acceleration, atmospheric jitter, and plume pogo states were subjected to the observability gramian condition test with both constant target angle and general target angle measurement models. In all cases, extremely large variations in the orders of magnitude of the eigenvalues of the observability gramian matrix M_o were observed, which would normally indicate unobservable states, which is a manifestation of numerics problems. In addition, both negative and complex eigenvalues were observed, . The inconclusive results are due to numerical precision problems in the software and hardware used to conduct the test.

Two approaches were attempted to improve the ill-conditioning of the state transition matrix Φ and reduce the variation in the orders of magnitudes and associated numerics difficulties. The first approach, reducing the sample period by 50 percent, had no significant impact. The second approach was to apply a scaling matrix to the state transition matrix prior to computing the system observability gramian matrix. This approach was successful in reducing the large order of magnitude variation and yielded all positive and real eigenvalues of the observability gramian matrix M_D. One could attempt to observe groupings of the eigenvalue magnitudes to gain insights into relative observability of the associated states, but definite conclusions cannot be drawn since the variation in order of magnitudes exceeds the reliable precision of the software and hardware with which the tests were conducted.

7.2.2 MSOFE Monte Carlo Analysis Results. Section 5.3.2 contains a detailed analysis of the results from testing four different filter configurations under varying conditions and in three different real-world environments. The primary objectives of the tests were threefold: first to determine the relative estimatability of atmospheric jitter and plume pogo individually; second to investigate the existence of interactions among the pogo and jitter states; and third to provide independent verification of the jitter and pogo models outside of the AFIT software.

Sections 5.3.2.3 and 5.3.2.4 address the first objective and Section 5.3.2.5 addresses the second objective. On an individual basis, either or both phenomena can be estimated if the filter model is properly tuned to reflect conditions in the real world. In general, the best filters were able to estimate jitter with an accuracy of $1\sigma = 0.25$ pixels, which is an improvement over the jitter itself, which was simulated with $\sigma_a = 0.45$ pixels. This conclusion is important in that it is contradictory to results obtained from the AFIT model in previous years and this year.

The various simulations also did not show any evidence of direct corruption of pogo on jitter estimation or vice versa. However, the data clearly indicated that jitter has the dominating effect on filter sensitivity. Filter models that included jitter estimation but not plume pogo were able to compensate for unmodeled pogo errors by attributing the motion to the jitter states. The converse case, assigning unmodeled jitter errors to plume pogo, was not successful. This was especially true for cases of lower pogo oscillation frequencies and larger magnitudes. The results described in Section 5.3.2.6 support the conclusion that the more difficult conditions to estimate are slower pogo frequencies and larger magnitudes of oscillation. Finally, analysis of the pogo statistics did show that pogo errors are impacted significantly by phase errors as well as magnitude estimation errors.

7.2.3 AFIT Elemental Filter Performance. This thesis successfully developed and analyzed the desired single filter that combined the independent "FLIR" and "center-of-mass" filters used in the previous three theses [8,9,14]. Chapter VI contains a detailed analysis of the performance exhibited by the 9-state filter. The overall performance of the filter was not as good as the performance demonstrated by the independent-filter configuration. It was originally hypothesized that incorporation of pogo and all other states into a single filter, which utilized both the FLIR and low-energy-laser measurements, should yield better performance. The degradation in performance, however, appears to reinforce some of the conclusions originally proposed by Rizzo four years ago [40]. The degradation may be due to either poor estimatability of the atmospheric jitter, interaction of the jitter model with the pogo states, or a combination of both.

Investigation of the jitter estimation showed that, unlike previous AFIT filters, the temporally averaged standard deviation in actual jitter estimation errors did not vary significantly when taken immediately before the update (t_{i+1}^-), and immediately after the update (t_i^+). However, the actual errors were of the same order as the jitter process uncertainty itself, indicating that the filter was not estimating jitter any better than if the jitter model states were not included in the filter. The difference in results between the MSOFE simulations and AFIT simulations (and improved jitter results in the AFIT filter) lead to speculation that there still may be a problem with how the jitter is modeled in the AFIT software.

Since the elemental filter was not studied under differing pogo conditions, conclusions of performance sensitivity to pogo corresponding to the MSOFE analysis were not obtained. The analysis of pogo estimation plots supported the conclusion drawn from the MSOFE analysis that accurate pogo estimation was highly dependent on the pogo phase as well as magnitude. For this reason, the pogo magnitude and velocity to which the filter is initialized affect the phase error and how quickly the filter recovers. Use of larger initial uncertainties and noise strengths in the filter stucture for the pogo states would speed recovery of the filter, and then the steady state values implemented in this simulation could be utilized for steady state tracking. Additionally, it is hypothesized that the higher the pogo frequency, the more quickly the filter will recover. This hypothesis is fully supported by the observation drawn from the MSOFE statistics that higher frequency pogo is easier to estimate.

Some of the most fruitful conclusions associated with the elemental filter are derived fr[illegible] analysis of the filter's estimation of [illegible] center-of-mass to centroid equilibrium point offset distance. The performance is much poorer than expected, especially considering that it should be a constant, and it is derived directly from the low-energy laser offset measurement by simply subtracting out the estimated pogo. Two major characteristics of the [illegible]set error statistics, a large bias in the mean, and noisy estimation values, yielded a plethora of potential explanations and a veritable bounty of recommendations for future research topics.

The remainder of the detailed elemental filter analysis was expended in a discussion of the possible problems in the Fortran code indicated by the large offset bias error and noisy measurements. The magnitude and direction of the bias were characteristic of the bias in the speckle return measurement model, so the appropriateness of the Doppler measurement model as implemented for this research is suspect. Additionally, the rapid changes in offset value are caused by a heavy reliance on accurate low-energy laser measurements, when a frame-by-frame analysis of LEL measurement data revealed somewhat less reliability than the programmed tuning parameters. This reliability of the offset measurement may be a residual of a suboptimal technique for intercepting the missile with the low-energy laser, either caused by poor sweep geometry or poor geometry with respect to the tracking window FOV.

The final analysis section of Chapter VI discussed several miscellaneous items dealing with the software itself. The conclusions drawn in that section focus on upgrading the software documentation and truth model to allow more reliable and realistic simulations for future researchers.

7.3 Recommendations

This section concludes the thesis with a number of suggestions for future research topics. Some of the advice is drawn directly from the performance analysis accomplished for this research. This type of guidance is intended to assist anyone who intends to continue from where this research was terminated. The other type of recommendations given near the end of this section are more general pointers that are intended to contribute to the general pursuit of the ballistic missile tracking problem. The Recommendations section is organized in a parallel fashion to the preceding Conclusions section. As with the previous section, recommendations have been generalized in the interest of succinctness. The reader is referred to the main body of the text for specifics.

7.3.1 Stochastic Observability. These tests were inconclusive chiefly because numerical precision limitations. It would not be advisable to attempt them again unless an alternate approach, perhaps with different algorithms, software, and or hardware are made available.

7.3.2 MSOFE Monte Carlo Analysis. The initial implementation of the ballistic filter tracking models was time consuming, however many of the hurdles have been overcome. The success of the MSOFE simulations indicates that it may be useful to continue its use to verify independently and/or support the results gained from simulations in the AFIT software. The original intent was eventually to perform simulations in MSOFE under the identical conditions as the AFIT simulations. This goal is still desirable. The MSOFE simulations should either be modified to reflect the current AFIT model, the AFIT model should be revised, or ideally some middle ground should be sought based on direction from the sponsor.

If MSOFE research is to be pursued any further, the next major step in augmenting the filter models would be to characterize the low-energy laser offset measurements and re-accomplish the sensitivity analysis for comparison to the AFIT simulations. Ultimately, it may be possible to utilize MSOFE to obtain approximate tuning parameters for the elemental filters.

Since MSOFE is currently not capable of simulating a multiple model adaptive algorithm, it can only serve to verify the individual elemental filters out of the filter bank. If a shell enabling MSOFE to be used in a MMAF simulation is ever developed, however, or if some other well-tested and documented filter simulation program is ever made available that has the capability to perform MMAF simulations, it should be investigated. Such software development is currently being pursued at AFIT.

7.3.3 AFIT Elemental Filter Research. The successful estimation of jitter using the MSOFE models is contradictory to the findings from AFIT simulations. In addition, since MSOFE did not indicate any jitter/pogo interactions, it would be useful to implement a similar series of tests using the AFIT software. Increased modularity of the AFIT software will make it easier to accomplish simulations using different filter and truth models

Causes for the poor estimation accuracy of the offset state should be pursued. The results of the Doppler model currently implemented should be compared to the results obtained by Herrera to determine its appropriateness for this simulation. Once the Doppler measurement model is verified to be properly utilized, an investigation should be accomplished to determine the appropriateness of the laser sweep routine and the FLIR FOV geometry in the presence of pogo. An alternate laser sweep geometry and/or alternate FOV geometry should be investigated to determine if improvement in the offset measurement is feasible. Two simple suggestions are simply to expand the sweep parameters (both angle and distance) and to enlarge the effective tracking window by either expanding its dimensions or implementing the DRFOV routine developed by Rizzo [40]. Whatever approach is pursued, the laser sweep routine should be restructured so that it never goes into an endless loop.

At this time, a parallel research effort toward more accurate tracking could be supported. One branch would be to pursue the single filter model with states defined as in the elemental filter developed in this thesis. The other viable approach is to continue the philosophy of the previous research and intentionally to separate problem into two separate filters, each based on the type of measurement available. Theoretically, the single filter approach should be ideal, since all states can interact with one another as they do in the real world. However, since it is evident that the offset state is corrupting the other states in the single filter approach, the pogo should be incorporated into the two-filter scheme and its performance compared to that of the single filter.

The problems causing degraded filter performance should be investigated prior to pursuing final tuning of elemental filters for a MMAF. The initialization procedures of the truth and filter models should be given special attention, since both the pogo initialization problem highlighted in Section 6.2.4 and the initial velocity transient in Section 6.2.2 may be repaired by the same software change. One possible avenue to pursue is to investigate the template shifting and re-acquisition routines to determine whether they need to be revised for the new model.

7.3.4 General Software Upgrades. There are three major revisions to the software which should be considered. The first revision is considered by this researcher to be an absolute necessity. The second revision is highly desirable and the third revision, the most challenging, also allows for the highest potential contribution the general tracking problem.

The first revision addresses the fact that other errors in the software may be embedded in the truth model or due to improper sign conventions in the FLIR plane y-axis. The entire code should be scrutinized for unnecessary or missing documentation or obsolete code. The internal documentation should be brought up to the present date, and some consideration should be given to a separate log or manual that contains specific hints not contained in the theses. As mentioned before, incorporation of some the features of MSOFE to enhance modularity would also make the code more comprehendible as well as more efficient to operate and debug.

The second revision involves modifying the simulation model so that the truth and filter state vectors coincide. They don't have to be exactly the same, but fundamental states, such as the target dynamics states, should define the same target in the same coordinate system. Careful consideration should be given to the enhanced correlation algorithm and associated template generation. Revision of the controller, statistics, and plotting routines should be fairly straightforward. If this revision is accomplished, the researcher should also seriously consider simultaneously modifying the controller to center the FOV on the missile hardbody and driving the controller with the newly defined target dynamics states for the center-of-mass.

The third revision is the most difficult but holds the most promise for flexibility and power in future research projects. Until now, the AFIT simulations have been coded from scratch in Fortran. Because of its longevity and widespread use, a ***substantial*** library of verified and trustworthy routines has been developed. However, the basic approach is still limited by the extremely labor-intensive requirement of manual coding, debugging, statistics calculations, plotting, ad infinitum. The advance of technology in both software and hardware has greatly simplified these time-consuming and error-prone tasks. It would be highly beneficial if the simulation could be reaccomplished in an alternate software package. One potential candidate would be Mathematica [52]. Mathematica is not only a well-integrated package that contains much more powerful mathematical manipulations and programming tools, but it would enable researchers to accomplish all the diverse operations performed for this thesis within a single package. There would be no requirement for a separate plotting routine or manually programming in statistics equations or even keeping track of matrix indices. Either discrete or continuous time models could be programmed directly in state-space form and analyzed symbolically for observability, controllability, etc; and statistics and plots could be created without any importing or exporting of data. As noted at the beginning of this "soapbox" lecture, such a major revision would be extremely challenging and would require many hours of verification and comparison to the current simulations to ensure correct operation. This writer holds the conviction that, in the long run, this recommendation bears the highest potential for advancement of the tracking problem and learning experience for future AFIT graduate students.

Appendix A

Data Processing Statistics Method

This Appendix explains how the statistics listed in the thesis and plotted in the succeeding appendices were determined. The equations for the statistics and the data used to plot them are discussed.

The performance of the Kalman filters used in this thesis is evaluated using multiple Monte Carlo runs (five for the AFIT simulations and 10 for MSOFE simulations). A Monte Carlo analysis involves collecting statistical information generated from simulating samples of stochastic processes [21]. Ten Monte Carlo runs are generally considered sufficient to converge to the actual statistics that would result from an infinite number of runs [9,24]. After collecting N samples of truth model and filter model data for each of N Monte Carlo runs, the true error statistics can be approximated by computing the sample mean error and error variance for the N runs. The sample mean error and error variance are computed by:

$$\bar{E}(t_i) = \frac{1}{N}\sum_{n=1}^{N}\left[x_{truth_n}(t_i) - \hat{x}_{filter_n}(t_i)\right] \tag{A.1}$$

$$\sigma^2(t_i) = \frac{1}{N-1}\sum_{n=1}^{N}\left[x_{truth_n}(t_i) - \hat{x}_{filter_n}(t_i)\right]^2 - \frac{N}{N-1}\bar{E}^2(t_i) \tag{A.2}$$

where

$\bar{E}(t_i)$ = sample mean of the error of interest at time t_i

$\sigma^2(t_i)$ = sample error variance at time t_i

$x_{truth_n}(t_i)$ = truth model value of the variable of interest at time t_i during simulation n

$\hat{x}_{filter_n}(t_i)$ = filter estimate of the variable of interest at time t_i during simulation n

N = number of Monte Carlo runs

The statistics are calculated before the measurement update at (t_i^-) and after the update at (t_i^+). In the performance plots displayed in Appendices B through E, the statistics at each instant in time are plotted together; that is, the statistics before and after the measurement update are plotted on the same time axis. They are reduced further to obtain average scalar values over the time of the run, by temporally averaging the mean error and standard deviation (σ) time histories from two seconds into the simulation until the end. The first two seconds are not used to ensure that the data reflects only steady state performance [9]. The errors are measured in units of pixels, where a pixel is 15 µrad on a side (approximately 3 meters at a distance of 200 km for the MSCFE analysis and 30 meters at a distance of 2,000 km for the elemental filter simulations).

Appendix B

Examples of Performance Plots

This appendix displays an example of the performance plots referenced throughout the thesis. An explanation of the plot components and their meaning is also given.

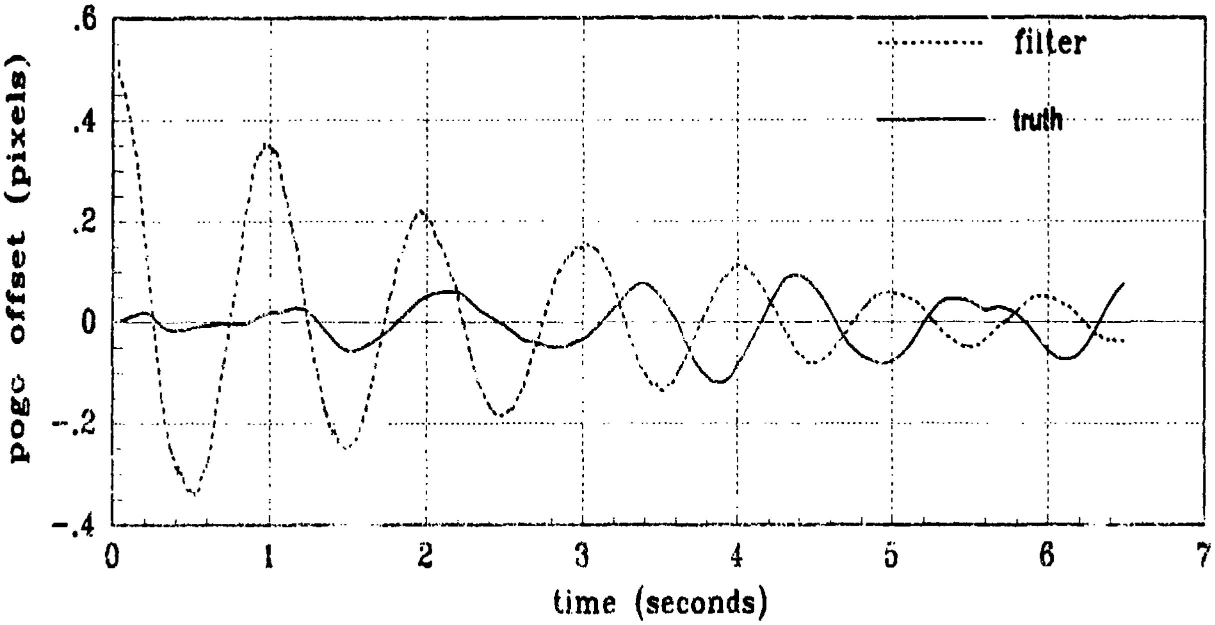
Two different types of data plots are presented in Appendices C through F to assess the performance of the center-of-mass filters employed in this thesis. The first type of plot, the state comparison plot shown in Figure B.1(a), provides a direct comparison of the filter estimated and true value of the state. For these plots, the ensemble average (over N Monte Carlo runs) of the true value of the state is shown as a solid line. The ensemble average value of the filter estimate at any instant in time is shown as a dashed line.

The second type of plot, the error statistics plot shown in Figure B.1(b), provides a measure of the tracking peformance. The plot shows the mean filter error, averaged over the N Monte Carlo runs at each instant in time, for a state or variable of interest. In addition, this type of plot displays the actual 1σ (standard deviation) centered on the mean, or mean ± 1σ curves. They are the two dotted lines that surround the mean curves. All the filters for this thesis were designed to assume zero mean errors in all states, so the filter computed estimate of standard deviation is plotted relative to the abscissa. The legend for the symbology in the error statistics plots is shown here.

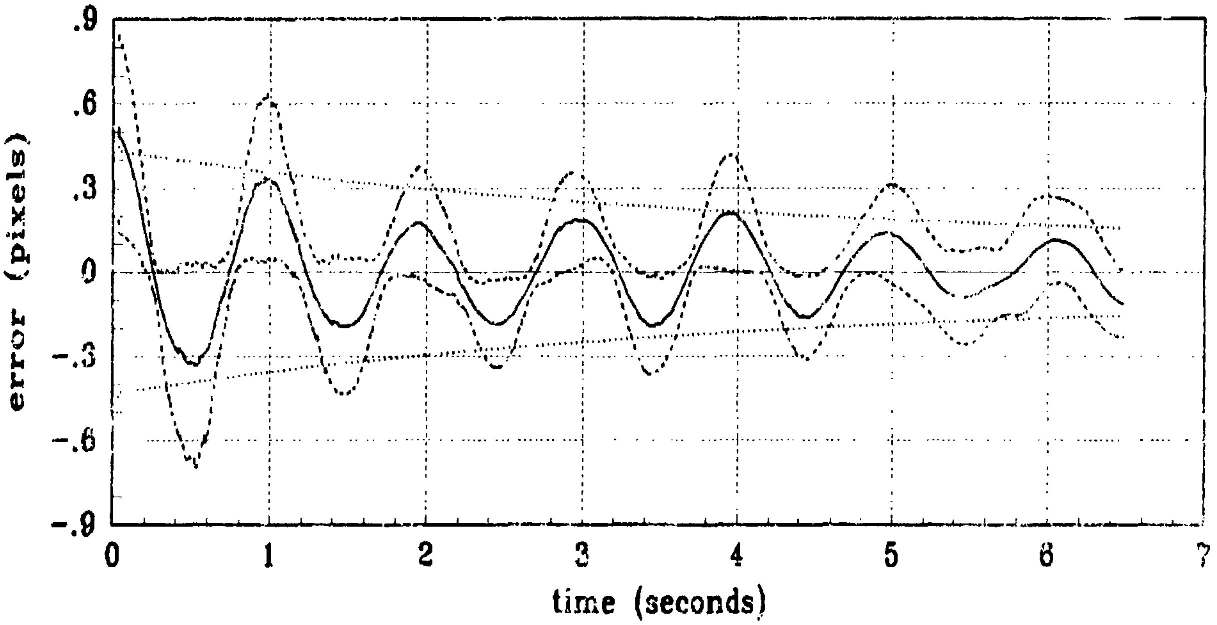
Mean Error ————————

Mean Error ± 1σ ……………………

zero ± Filter Computed 1σ - - - - - - - - - - - -



(a) FILTER vs TRUE STATE



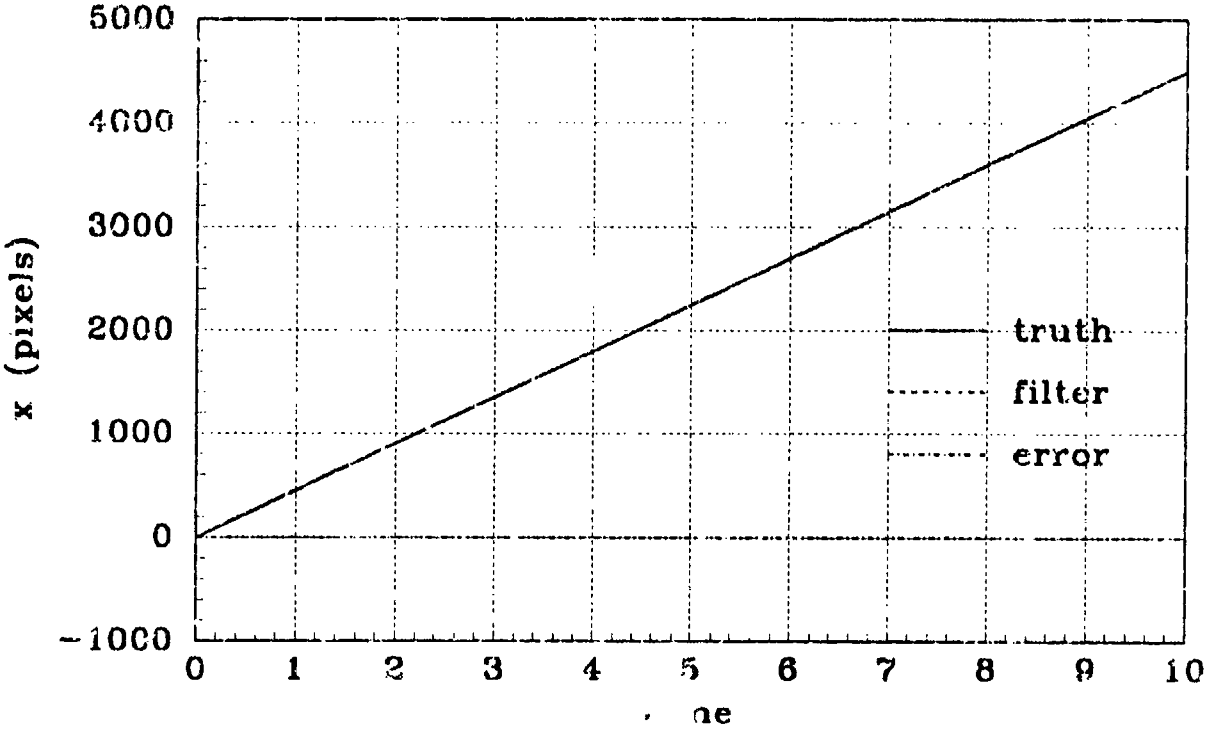
(b) FILTER vs ACTUAL ERROR (STATE)

Figure B.1 Example of State Plot and Error Statistics Plot

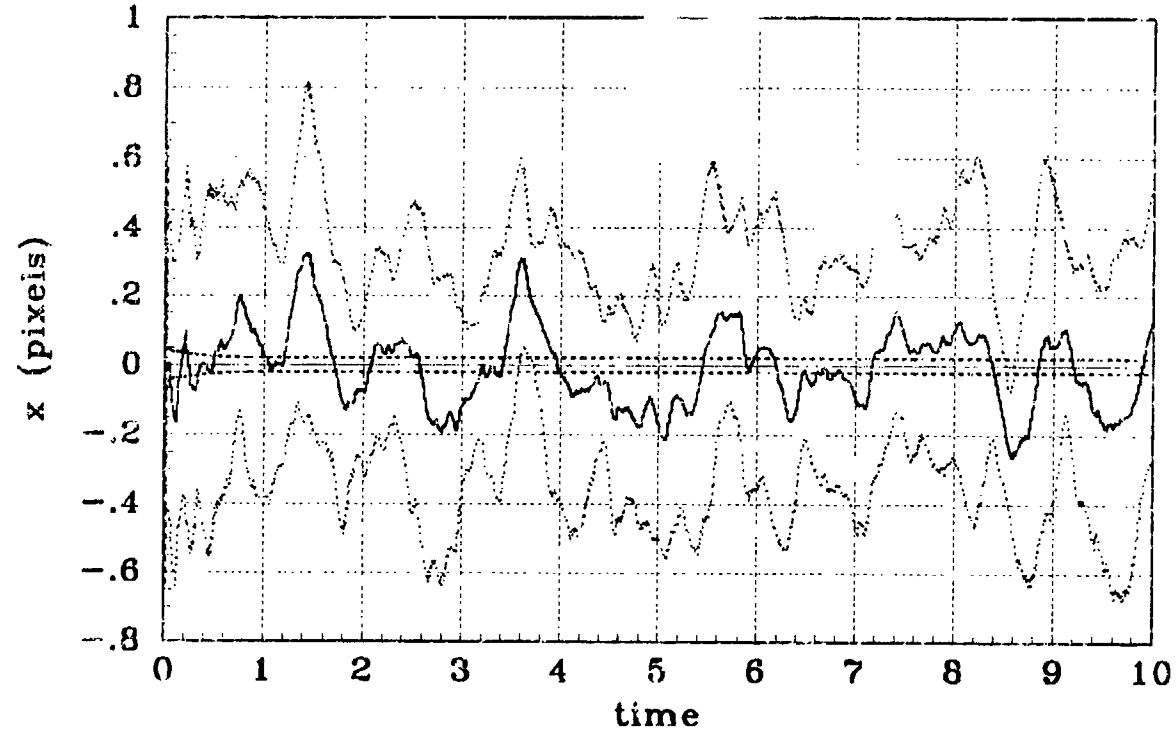
Appendix C

MSOFE Analysis of Filter Models Using Linear Update

This Appendix contains the average states and error statistic plots for the MSOFE simulations listed in Table 5.2. Data collection and symbology depicted in the two types of plots contained in this appendix are explained in Appendices A and B. The state comparison plots show the ensemble average truth state over the 10 Monte Carlo runs compared to the same statistic for the filter estimate. The error statistics plots represent the error mean and mean ±1 standard deviation values in pixels (or pixels/second for velocity) of the errors between the filter-estimated and true state.

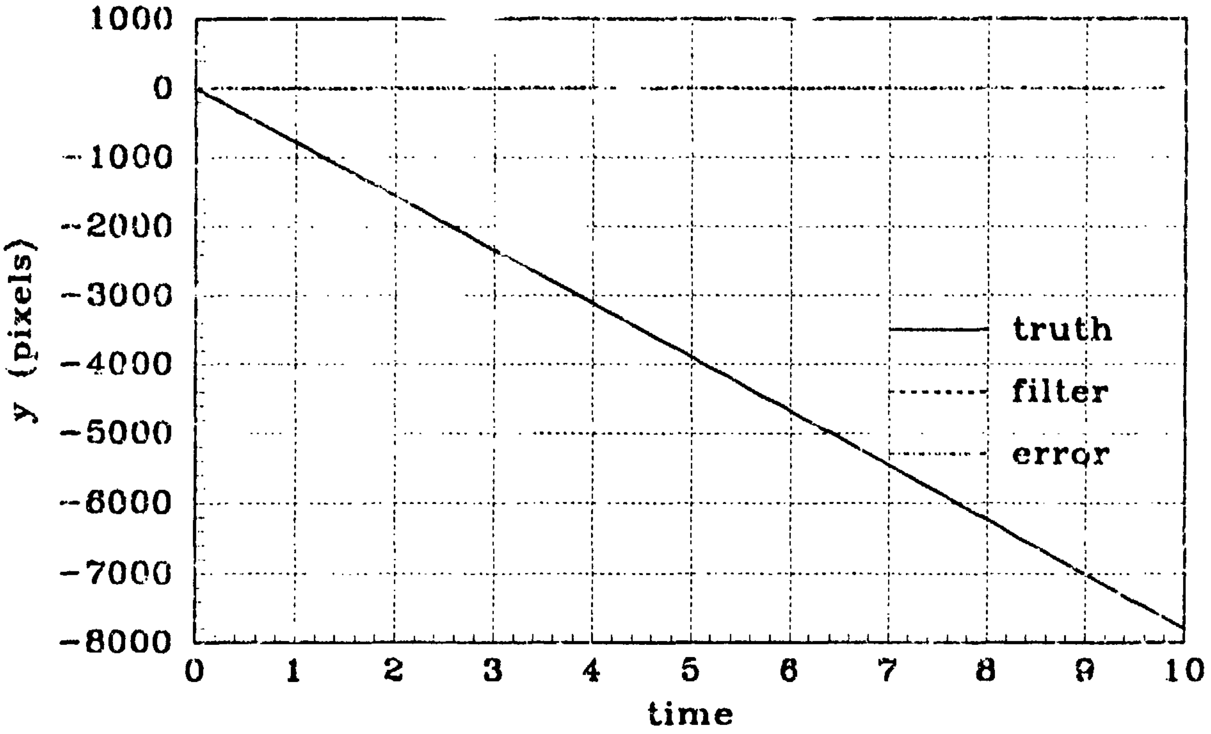


(a) X POSITION – Filter [illegible], linear

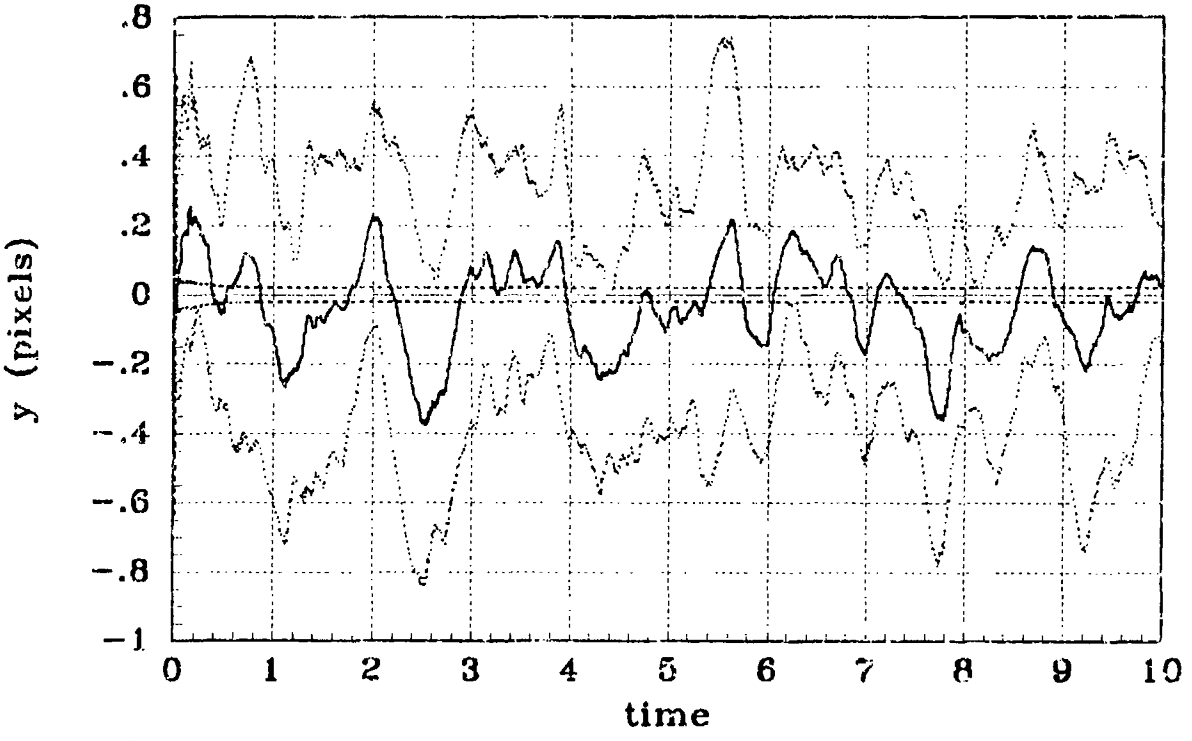


(b) X POSITION STATISTICS – Filter #2, linear

Figure C.1 Simulation #2, θ = 60°, X-Position

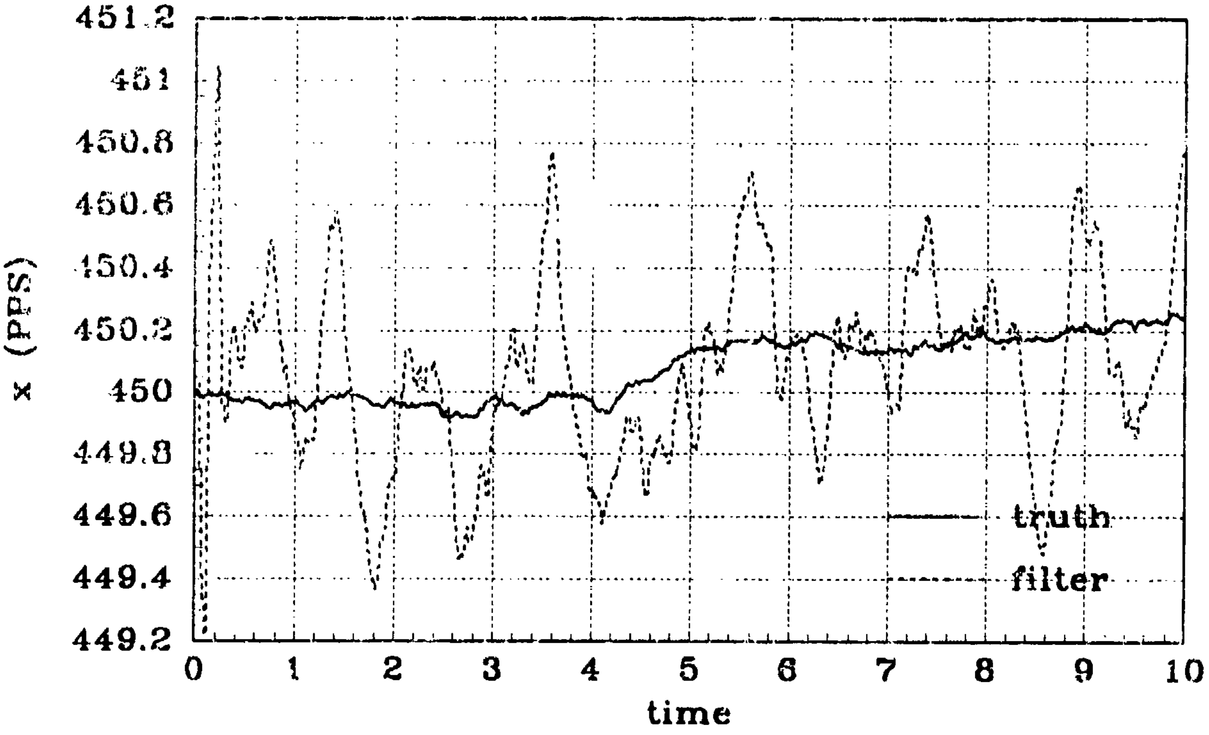


(a) Y POSITION - Filter #2, linear

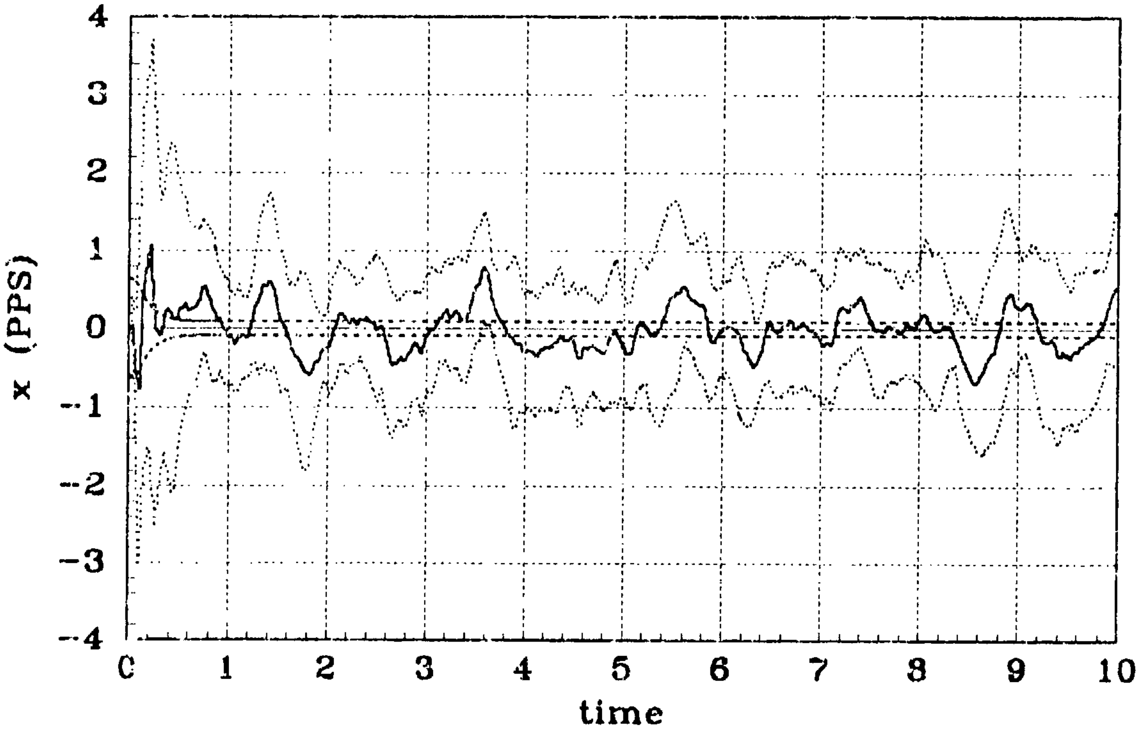


(b) Y POSITION STATISTICS - Filter #2, linear

Figure C.2 Simulation #2, θ = 60°, Y-Position

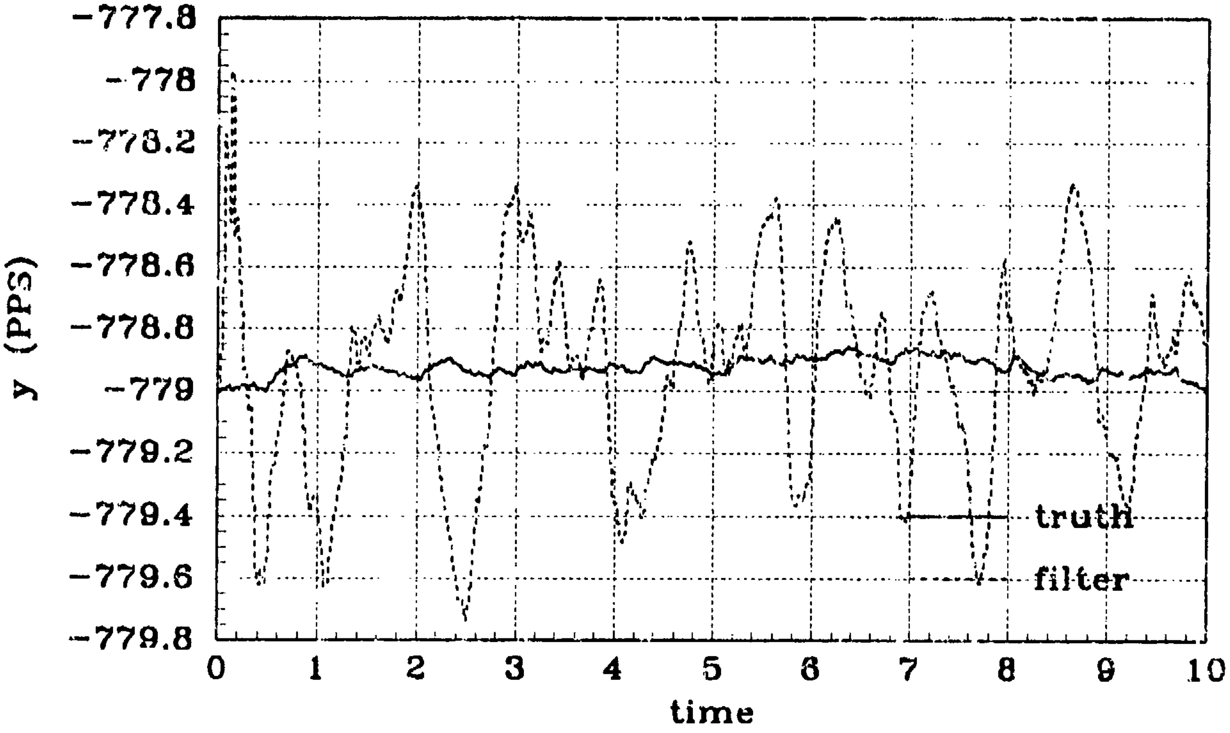


(a) X VELOCITY – Filter #2, linear

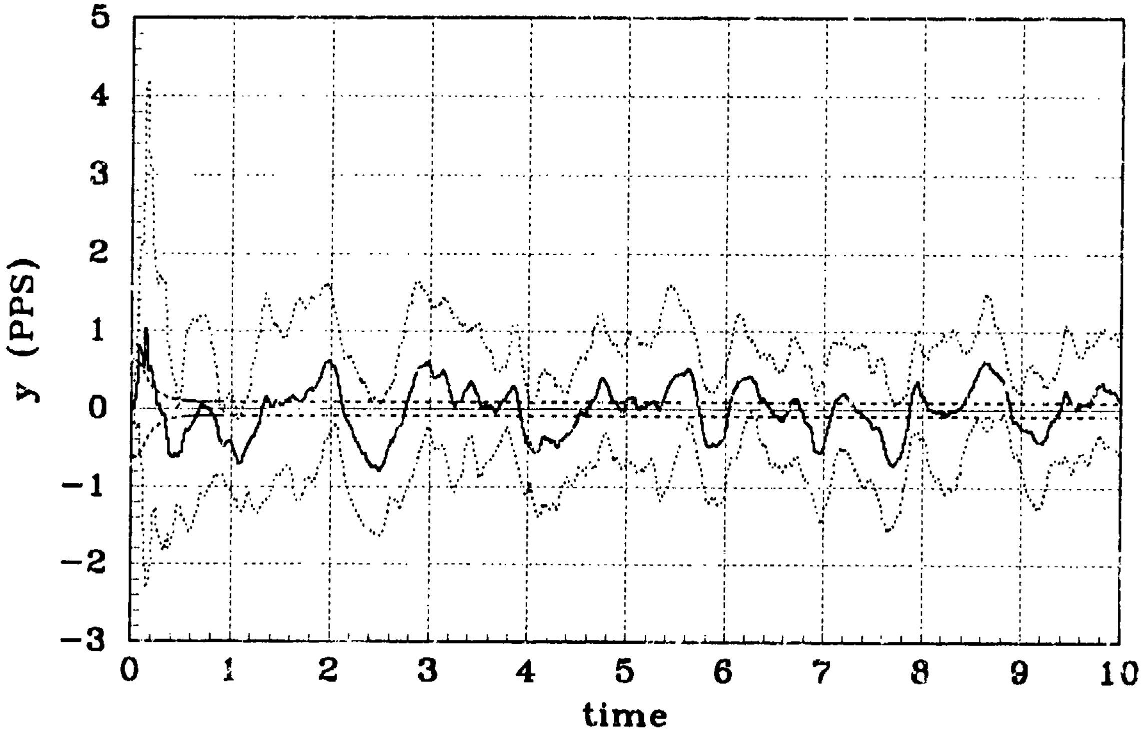


(b) X VELOCITY STATISTICS – Filter #2, linear

Figure C.3 Simulation #2, θ = 60°, X-Velocity

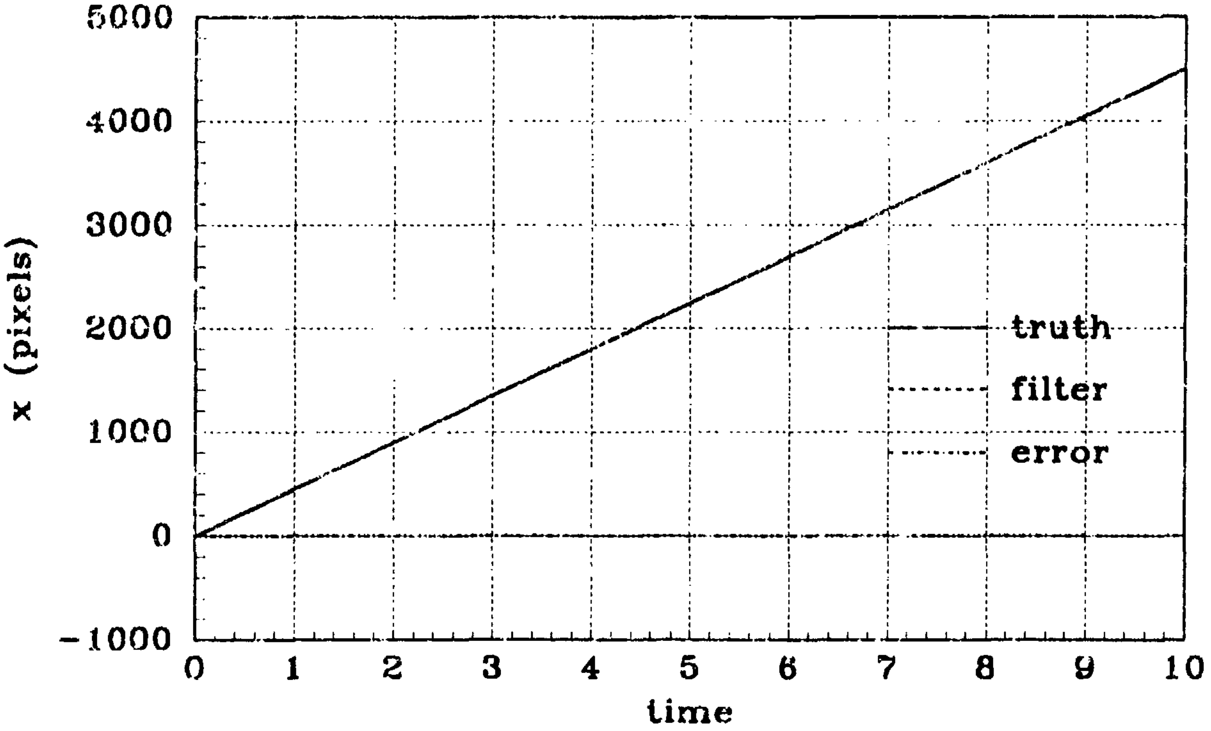


(a) Y VELOCITY – Filter #2, linear

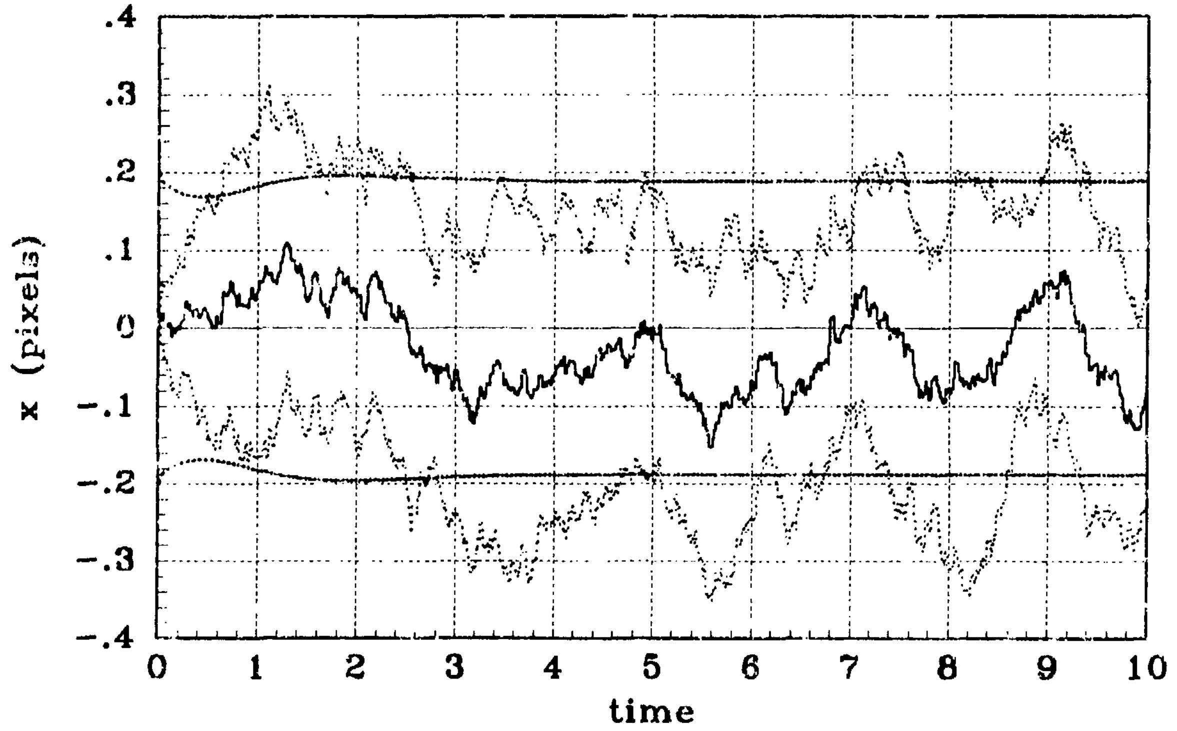


(b) Y VELOCITY STATISTICS – Filter #2, linear

Figure C.4 Simulation #2, θ = 60°, Y-Velocity

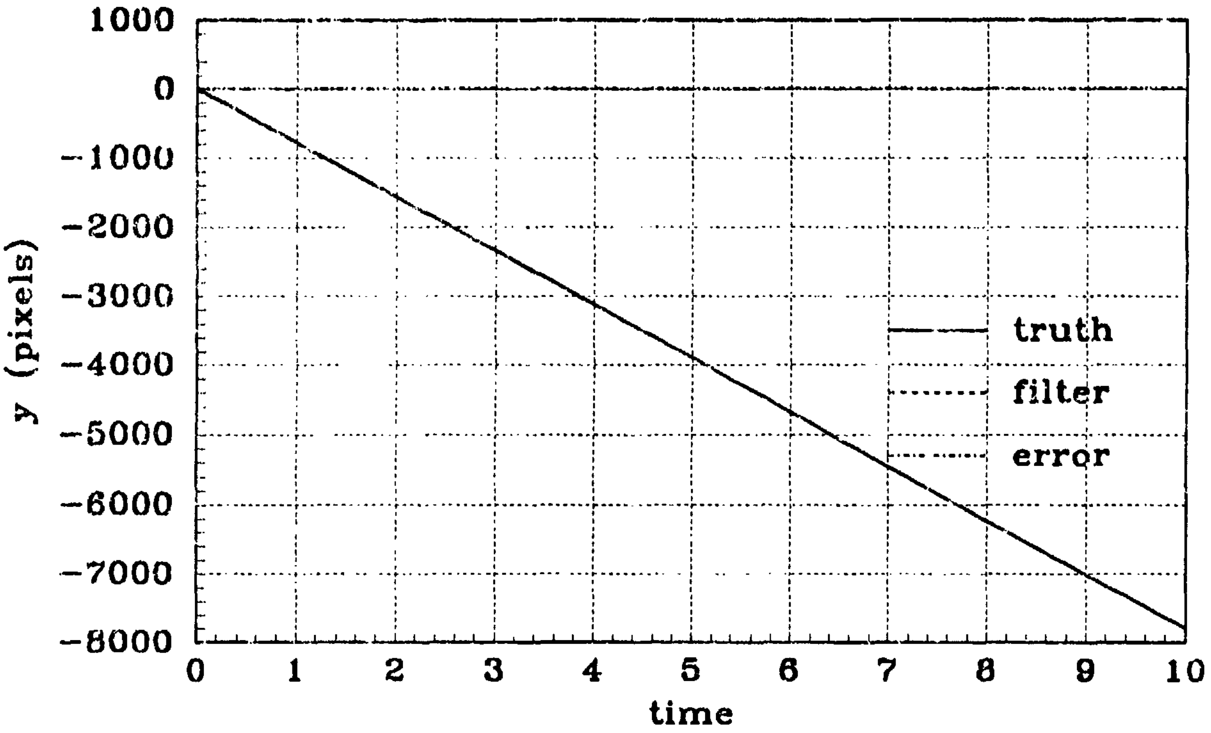


(a) X POSITION – Filter #4, linear

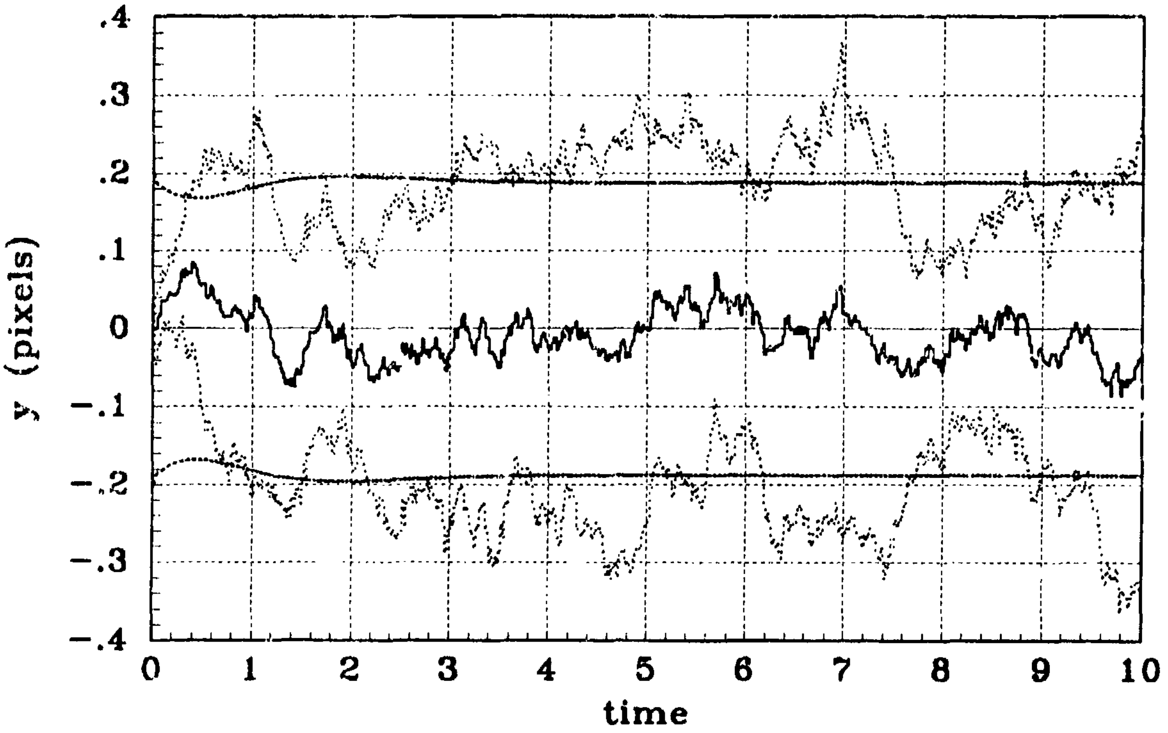


(b) X POSITION STATISTICS -- Filter #4, linear

Figure C.5 Simulation #4, θ = 60°, X-Position

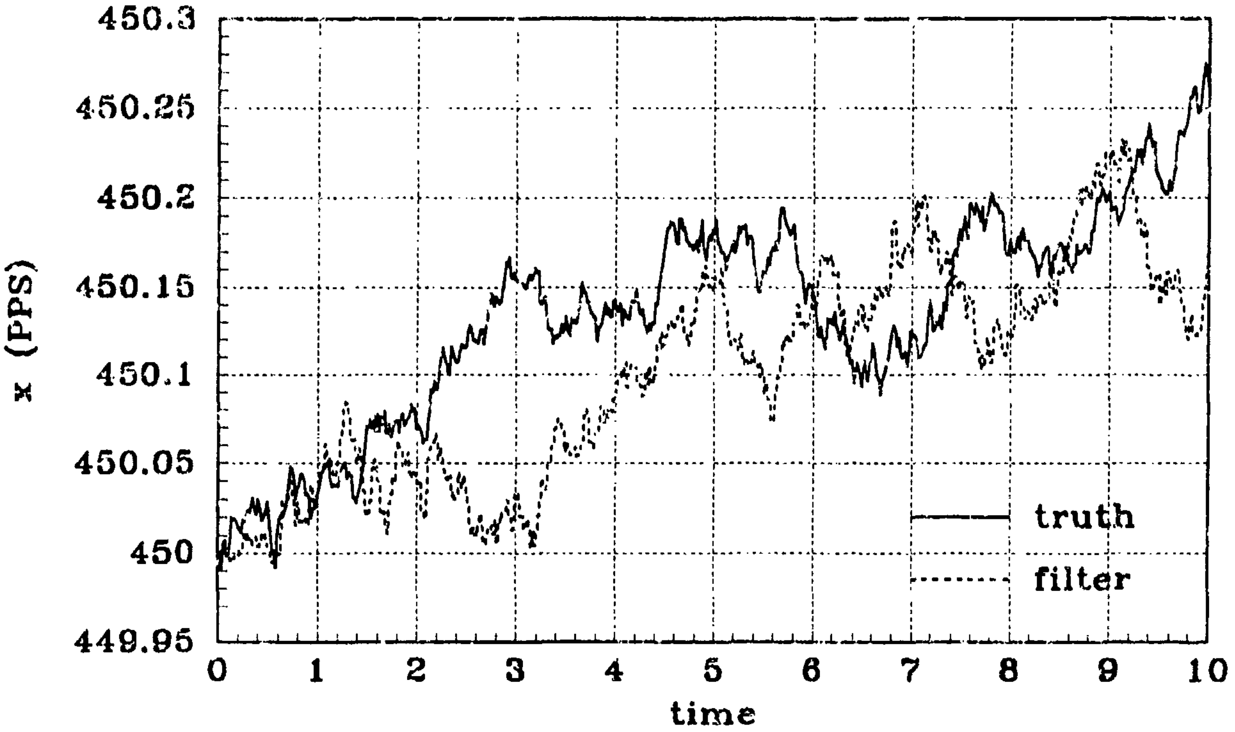


(a) Y POSITION – Filter #4, linear

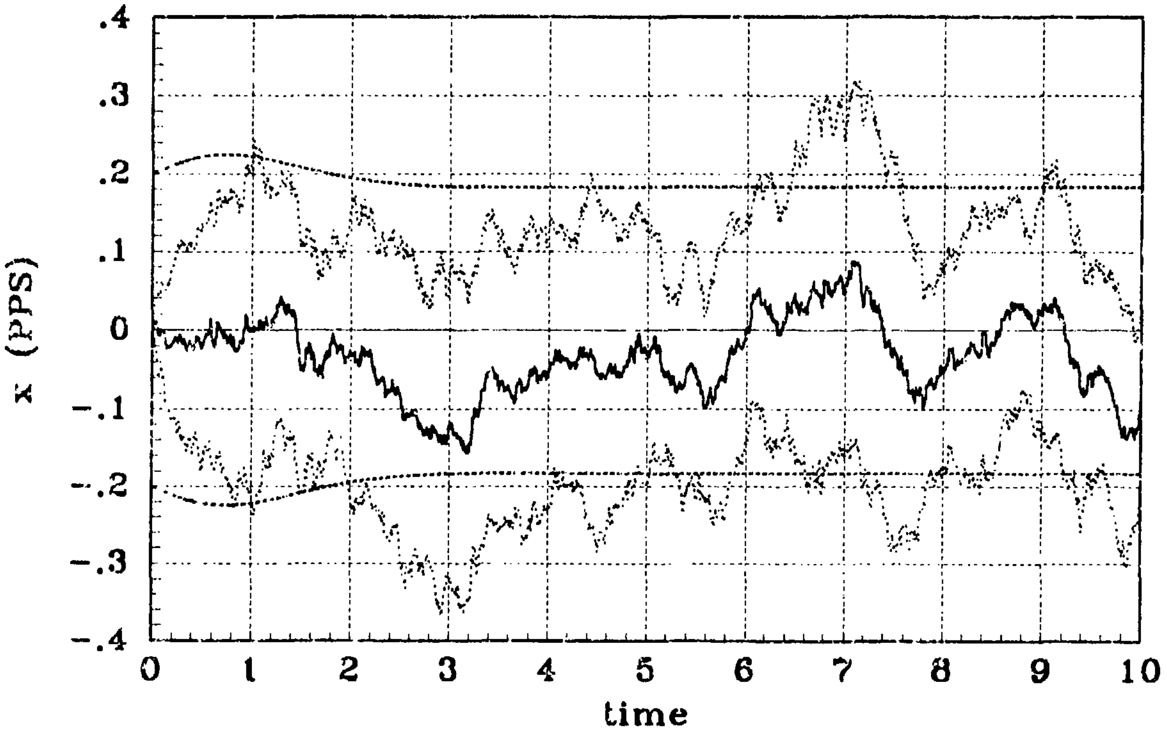


(b) Y POSITION STATISTICS – Filter #4, linear

Figure C.6 Simulation #4, θ = 60°, Y-Position

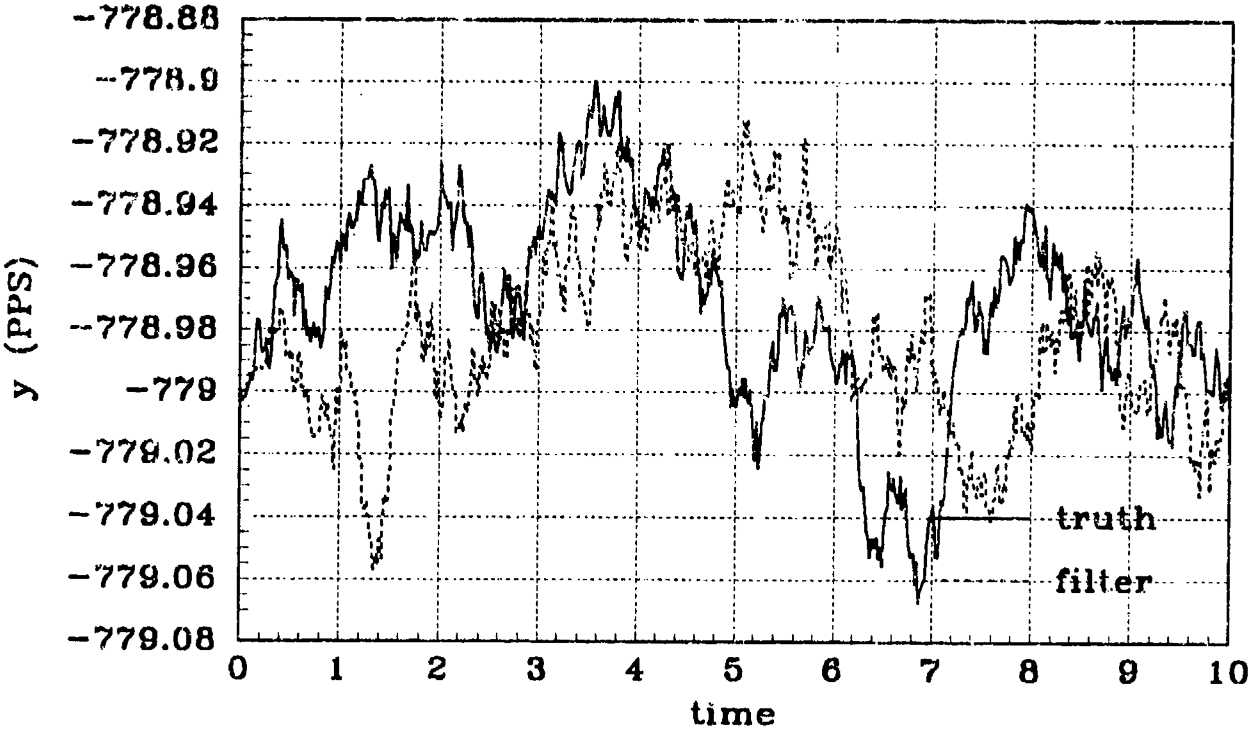


(a) X VELOCITY - Filter #4, linear

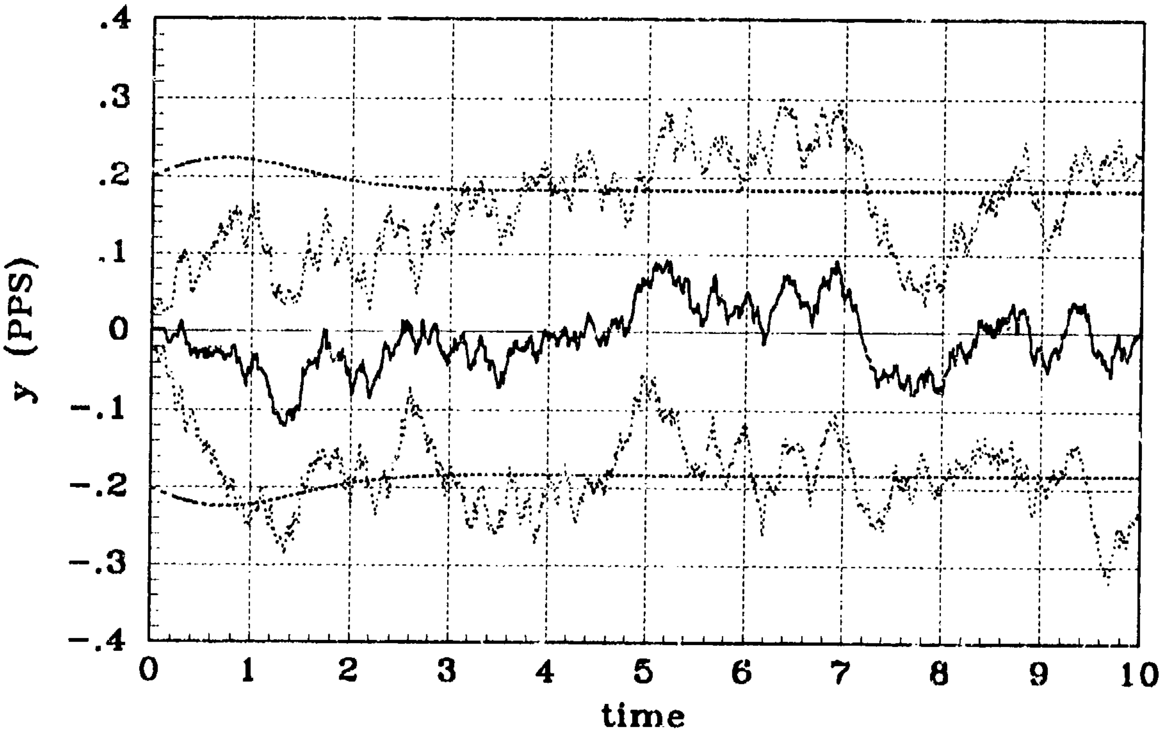


(b) X VELOCITY STATISTICS - Filter #4, linear

Figure C.7 Simulation #4, θ = 60°, X-Velocity

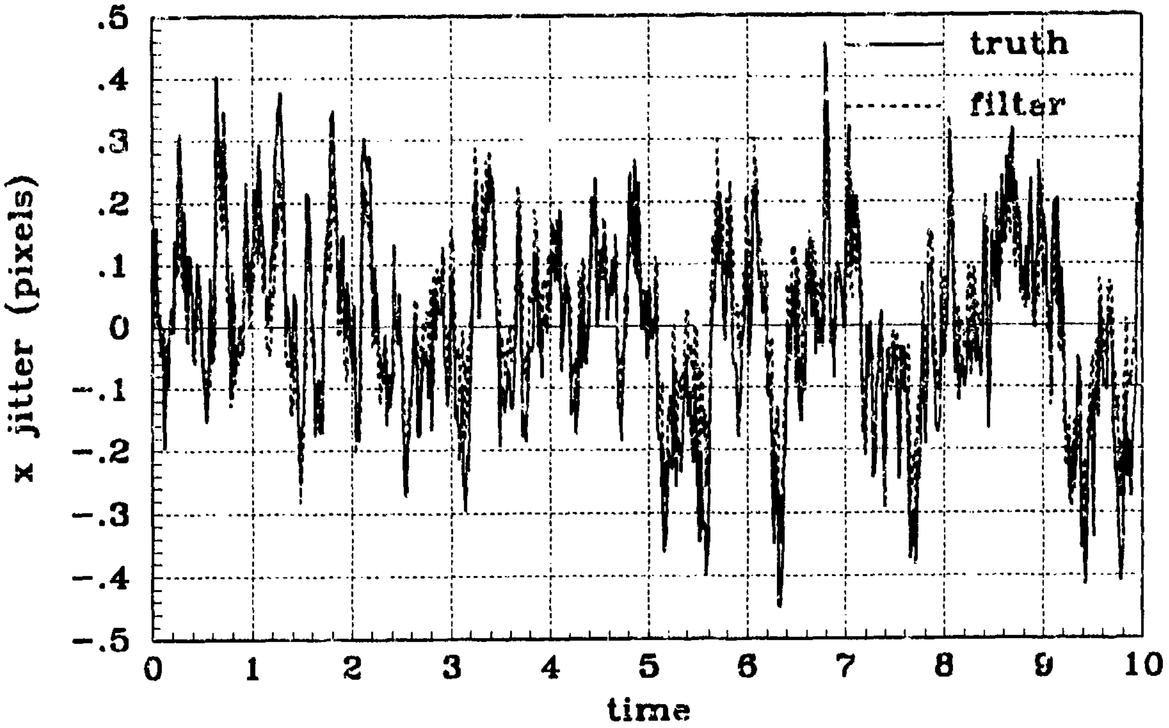


(a) Y VELOCITY - Filter #4, linear

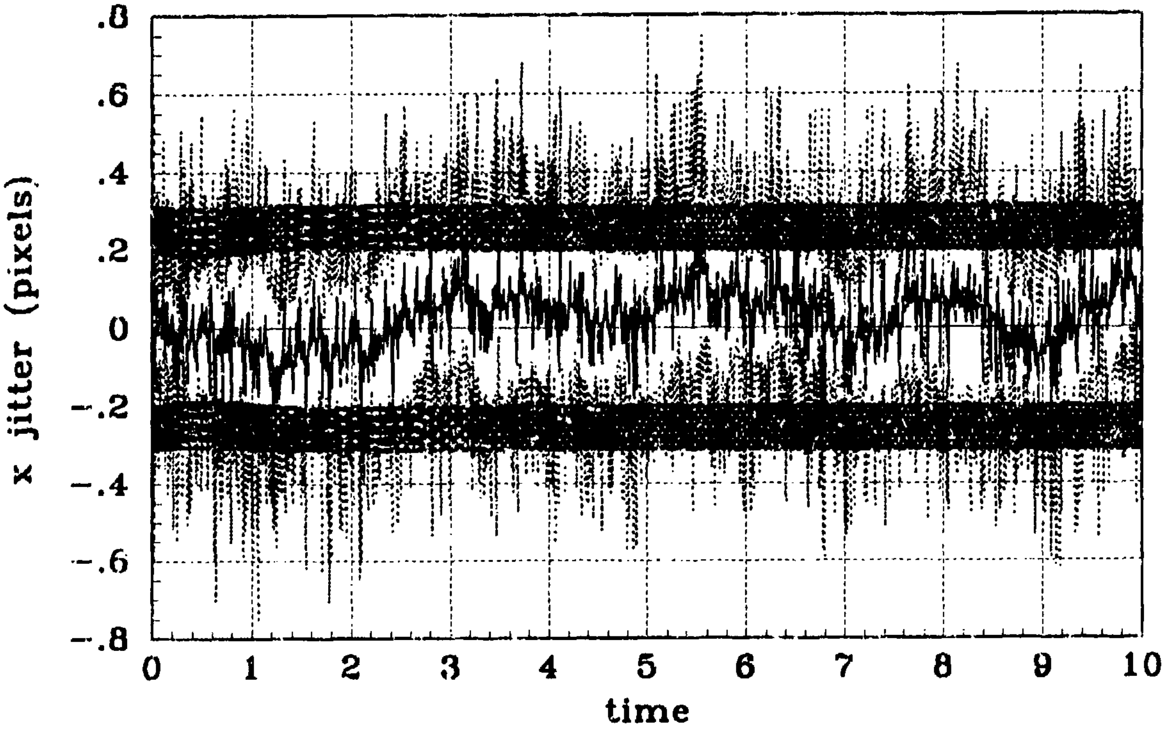


(b) Y VELOCITY STATISTICS - Filter #4, linear

Figure C.8 Simulation #4, θ = 60°, Y-Velocity

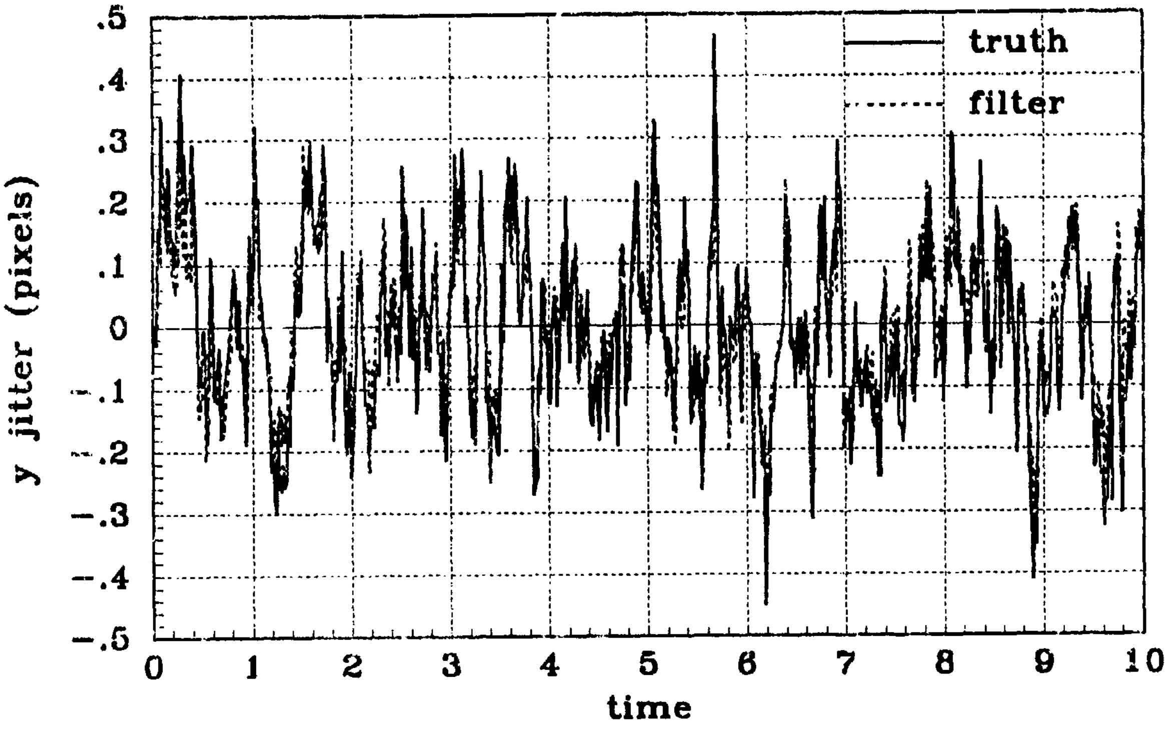


(a) X JITTER - Filter #4, linear

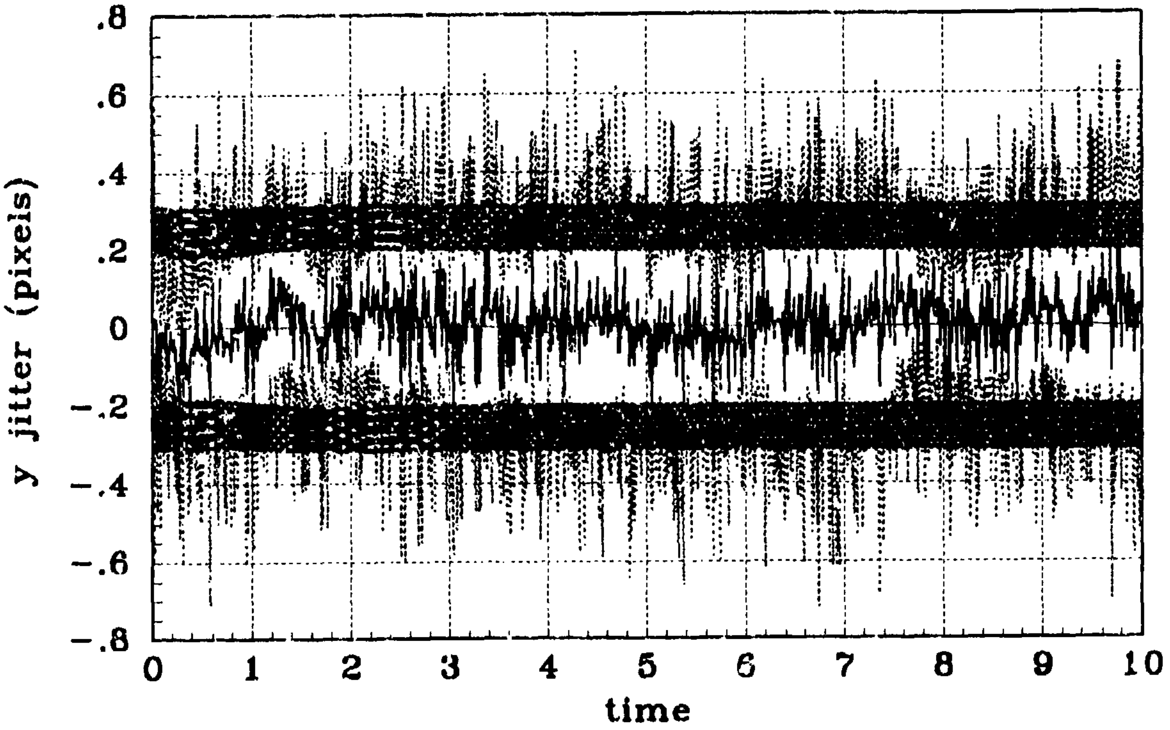


(b) X JITTER STATISTICS - Filter #4, linear

Figure C.9 Simulation #4, θ = 60°, X-Atmospheric Jitter

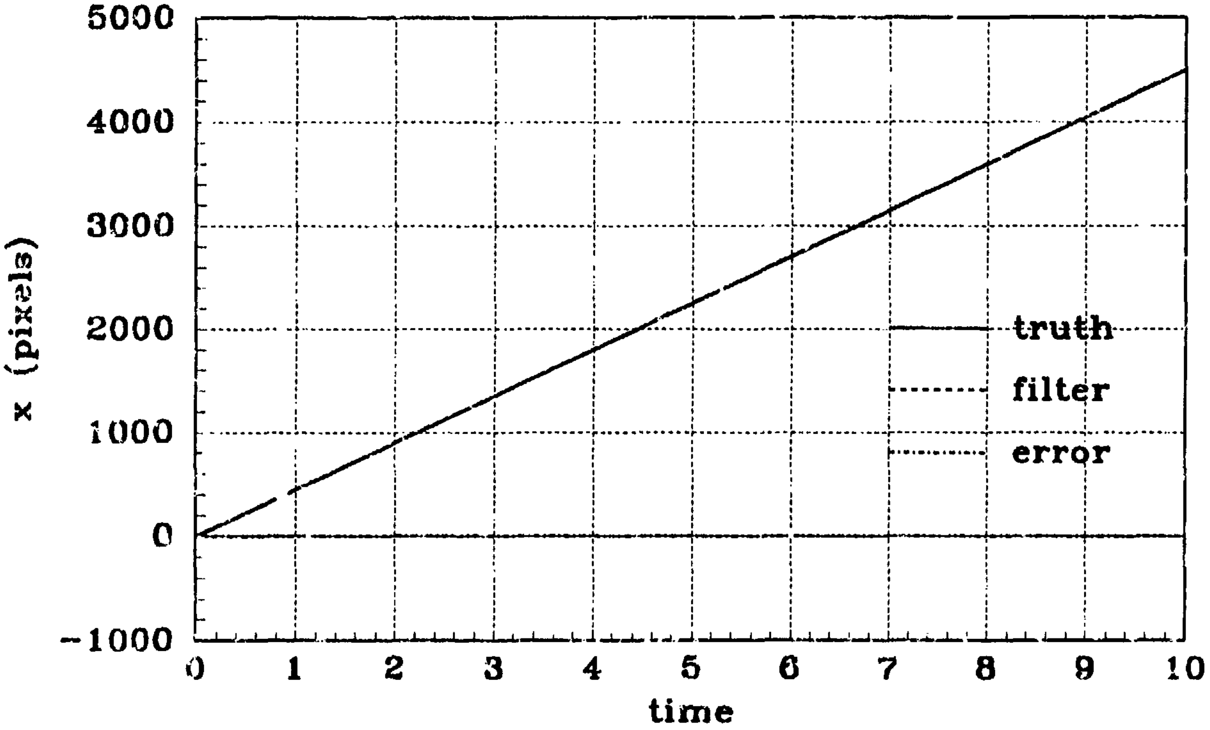


(a) Y JITTER - Filter #4, linear

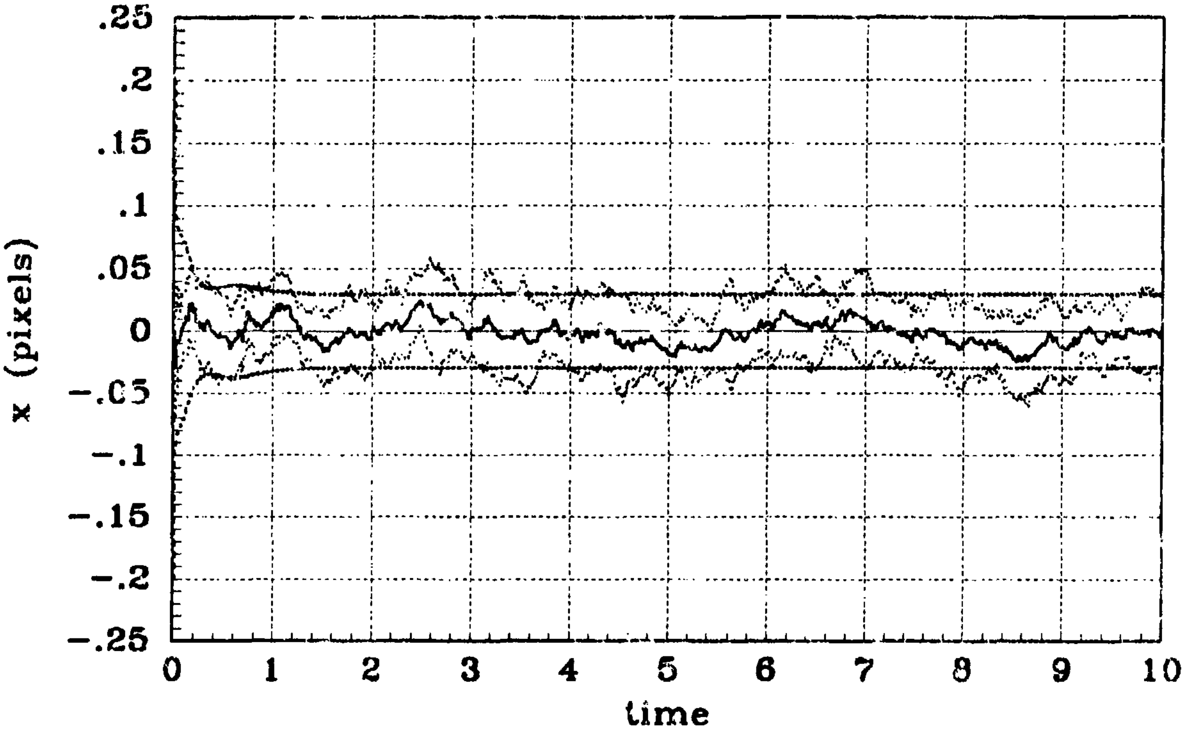


(b) Y JITTER STATISTICS - Filter #4, linear

Figure C.10 Simulation #4, θ = 60°, Y-Atmospheric Jitter

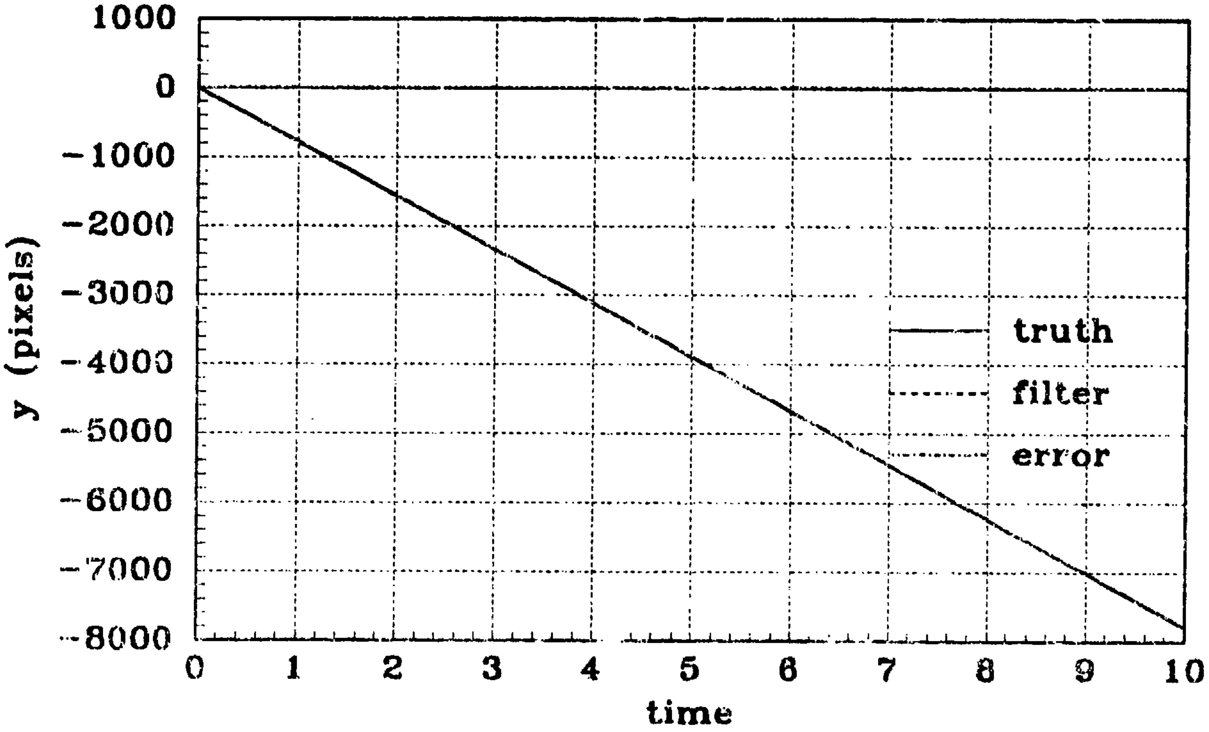


(a) X POSITION − Filter #6, linear

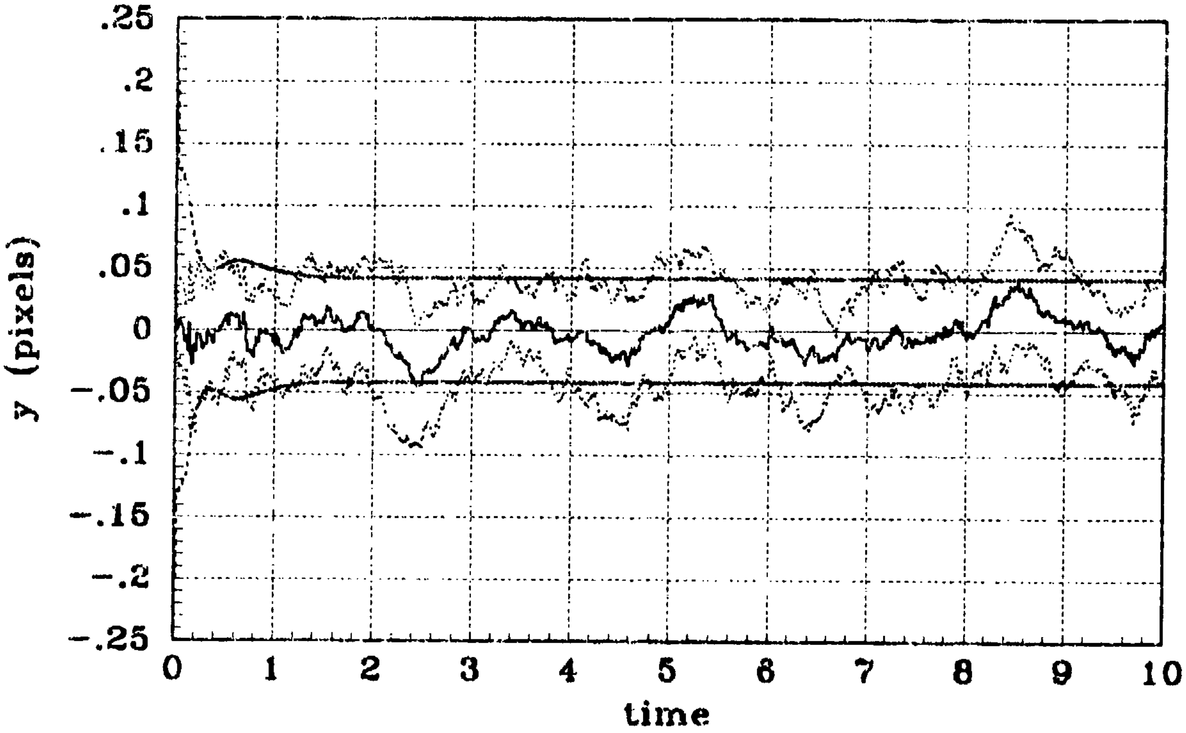


(b) X POSITION STATISTICS − Filter #6, linear

Figure C.11 Simulation #6, $\theta = 60°$, X-Position

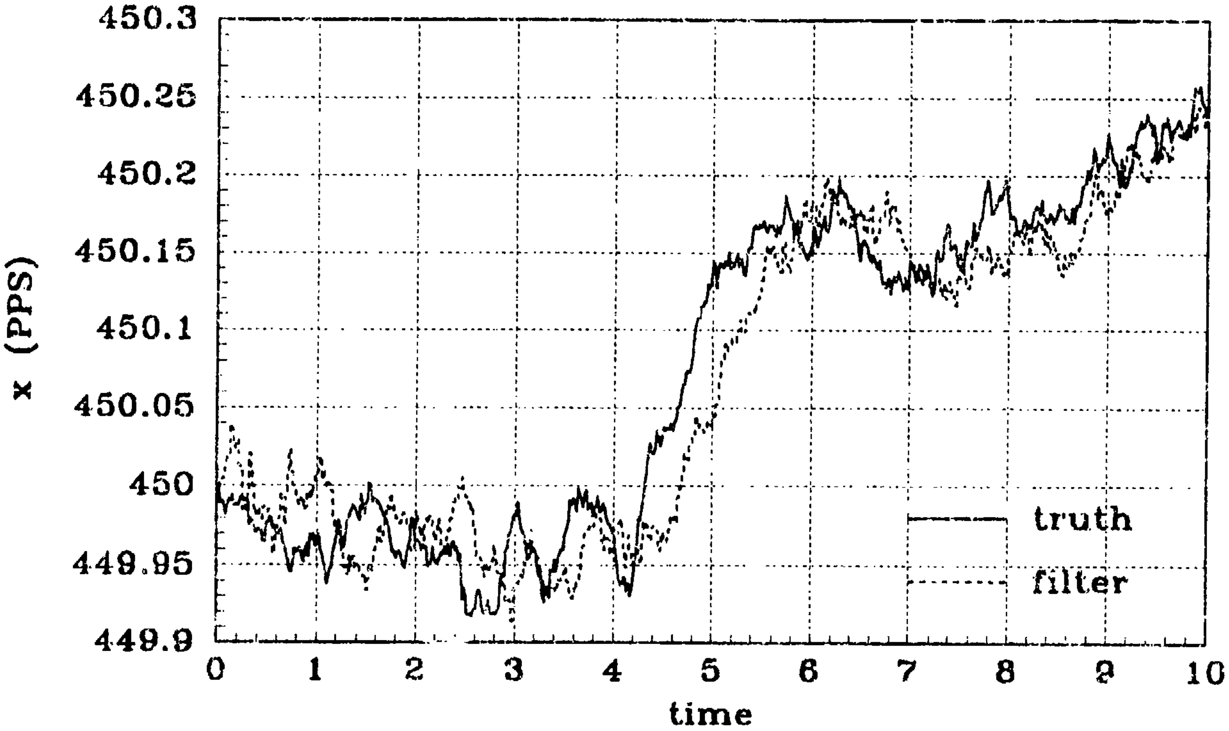


(a) Y POSITION – Filter #6, linear

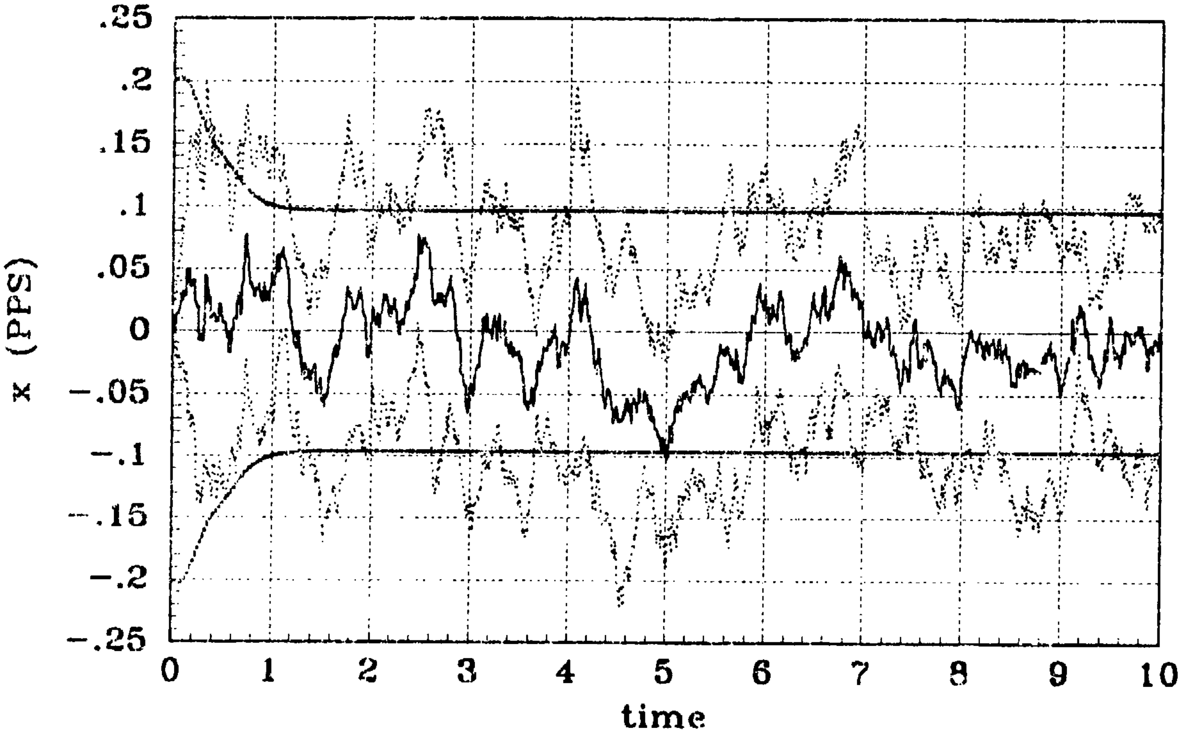


(b) Y POSITION STATISTICS – Filter #6, linear

Figure C.12 Simulation #6, θ = 60°, Y-Position

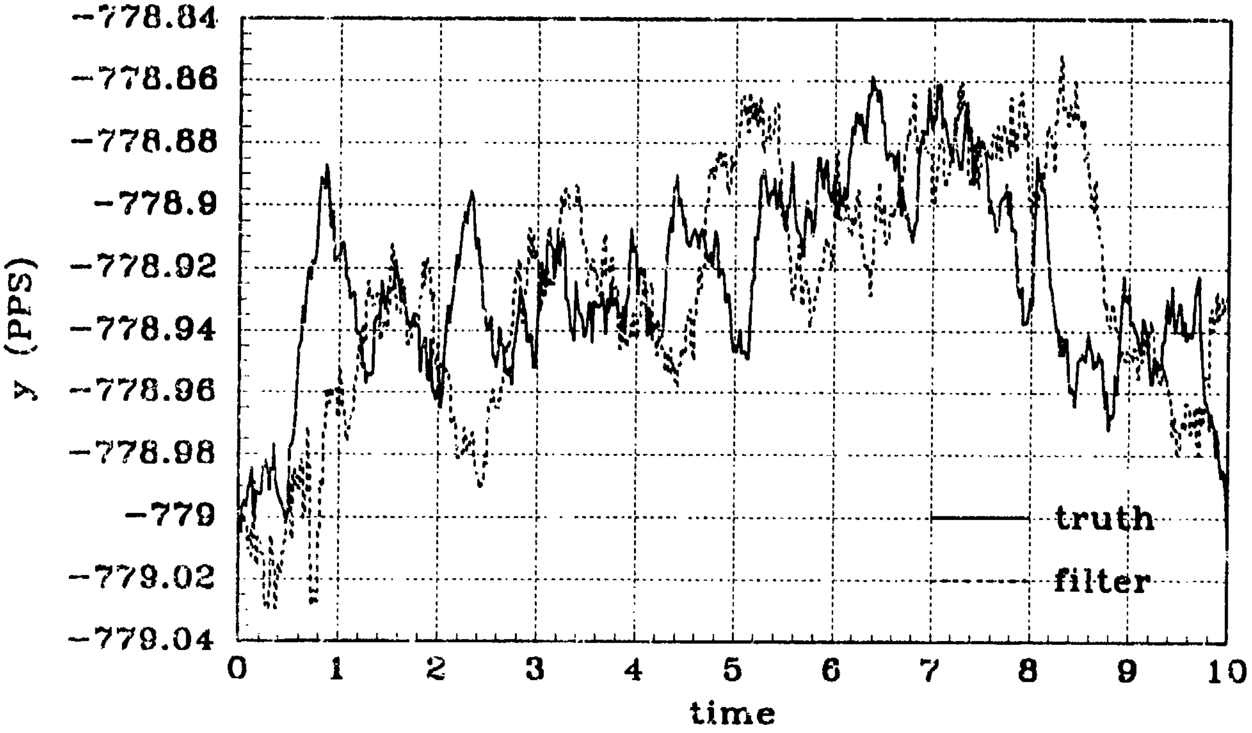


(a) X VELOCITY – Filter #6, linear

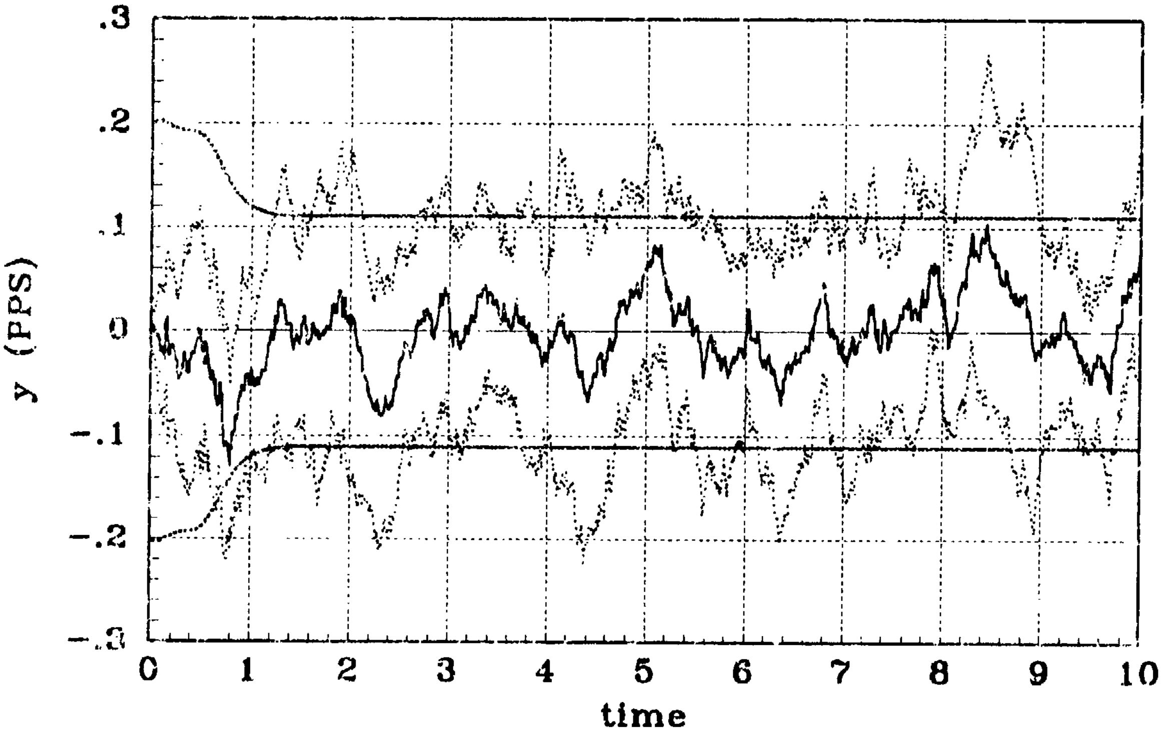


(b) X VELOCITY STATISTICS – Filter #6, linear

Figure C.13 Simulation #6, θ = 60°, X-Velocity

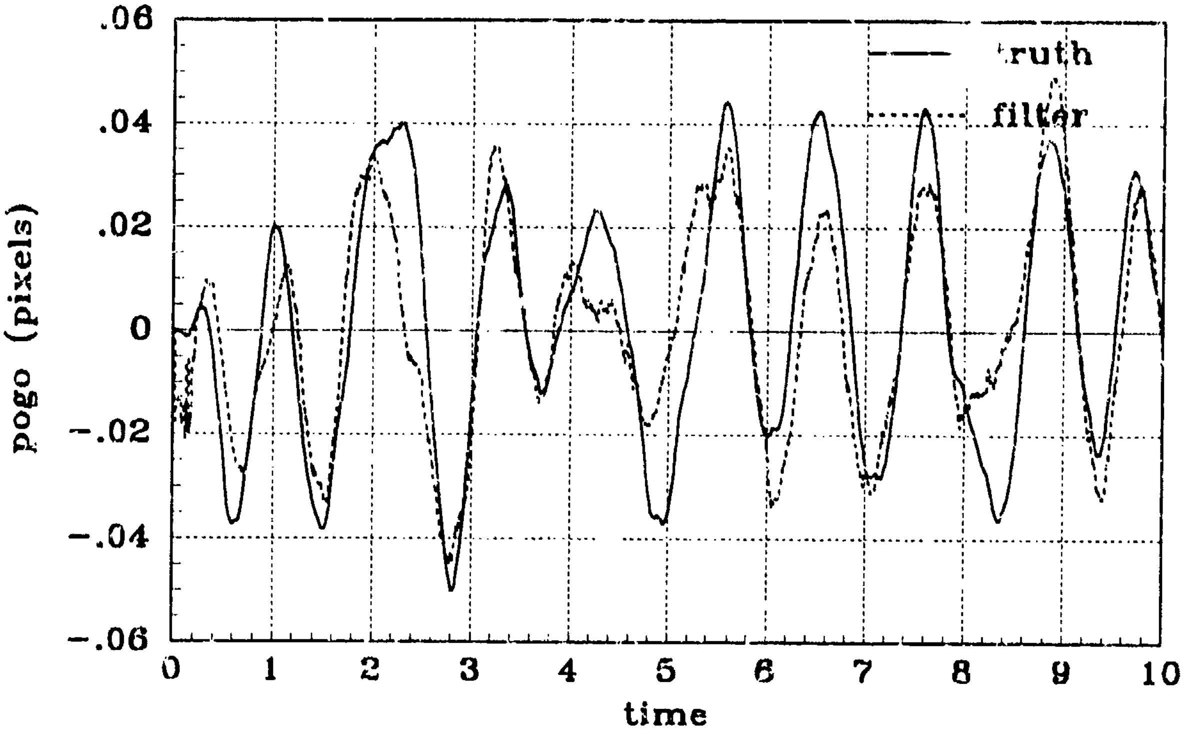


(a) Y VELOCITY - Filter #6, linear

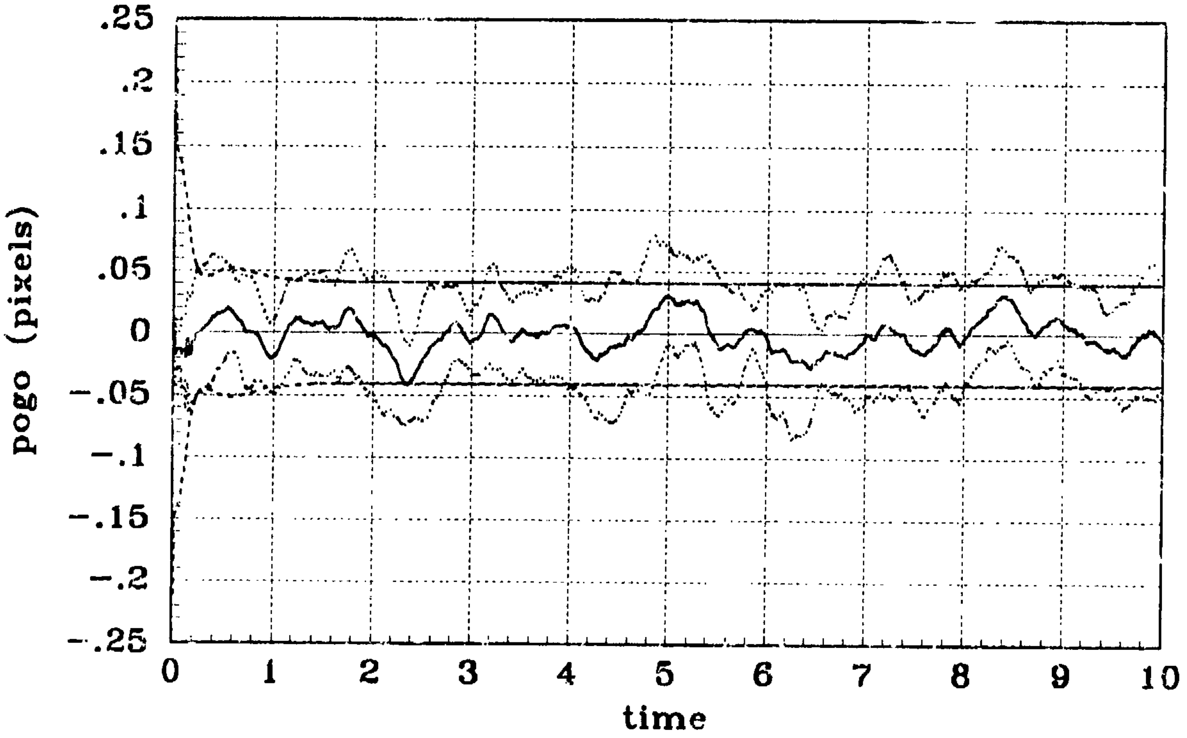


(b) Y VELOCITY STATISTICS - Filter #6, linear

Figure C.14 Simulation #6, θ = 60°, Y-Velocity

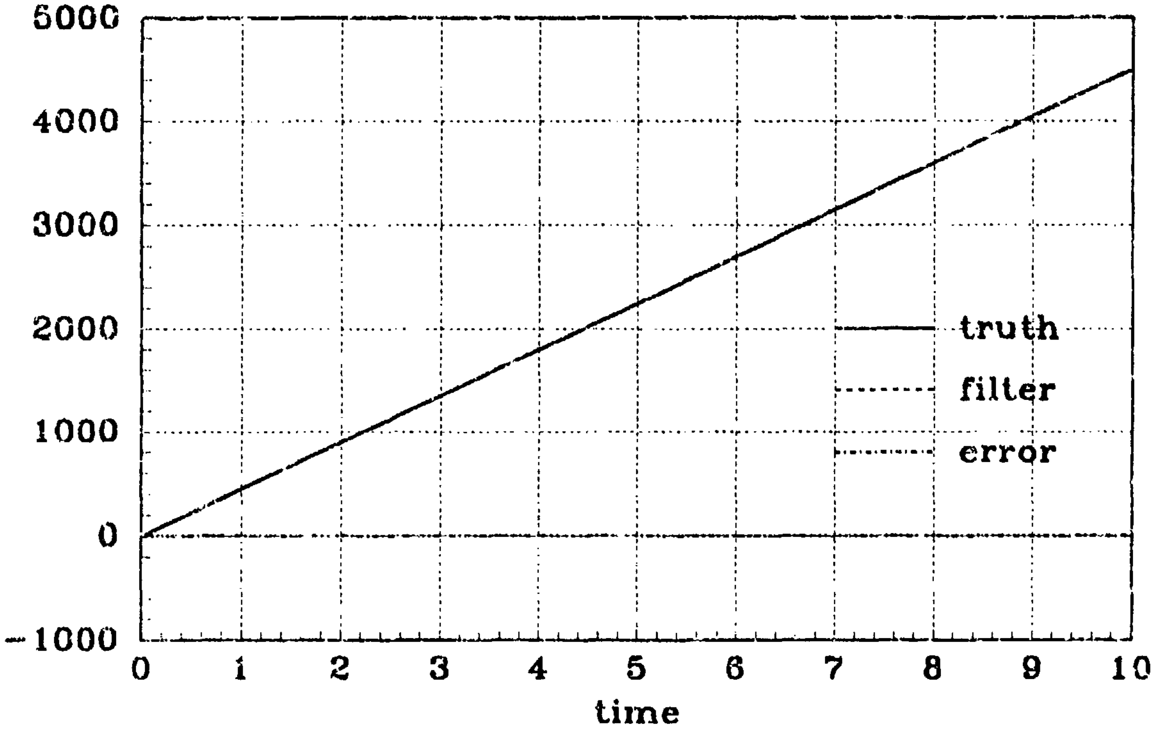


(a) POGO OFFSET – Filter #6, linear

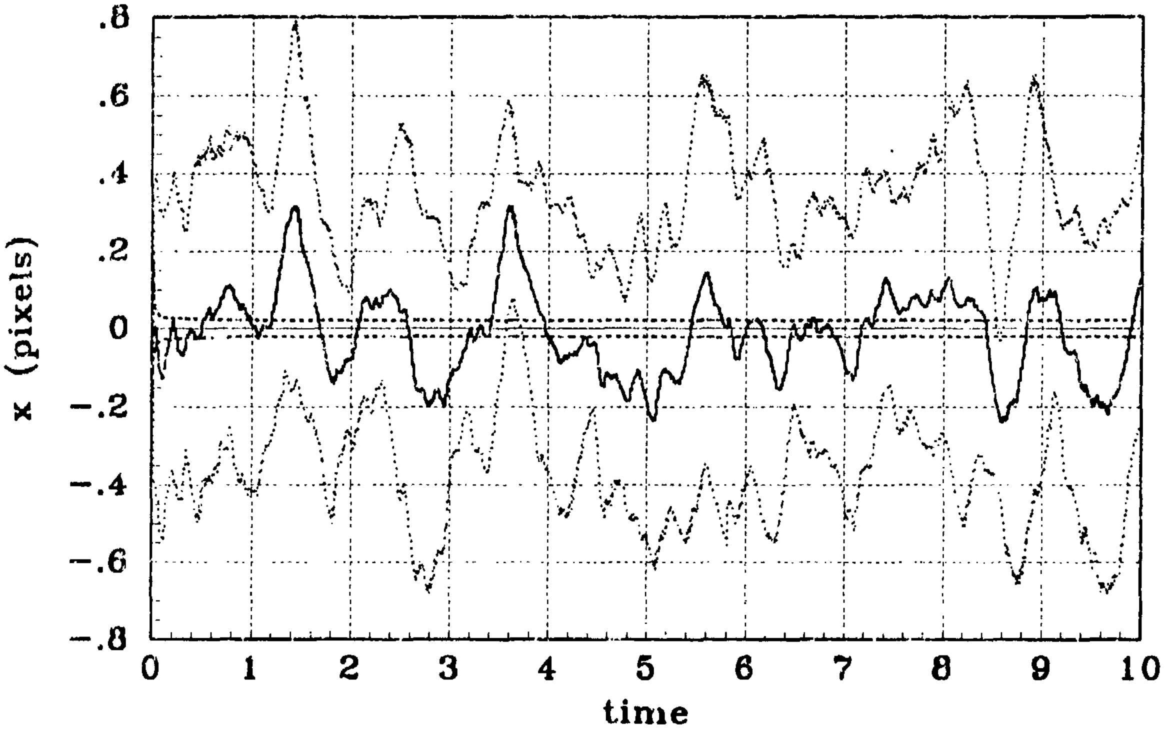


(b) POGO OFFSET STATISTICS – Filter #6, linear

Figure C.15 Simulation #6, θ = 60°, Pogo Offset

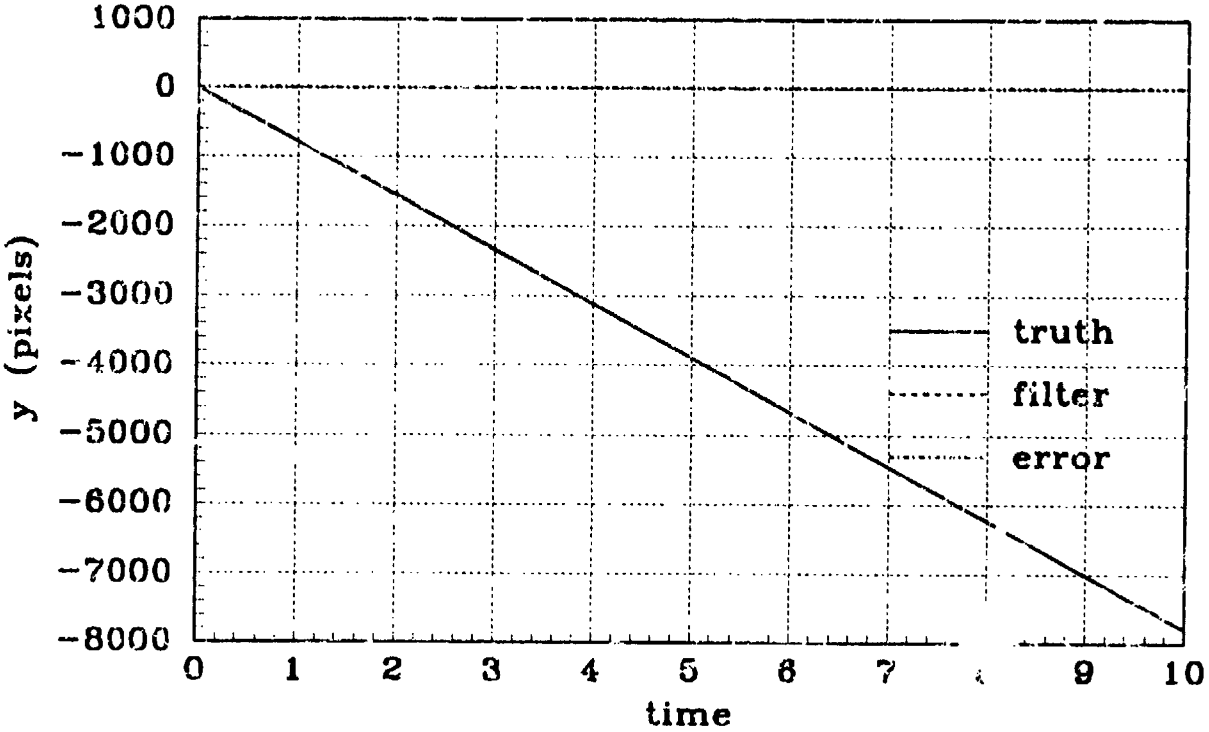


(a) X POSITION – Filter #7, linear

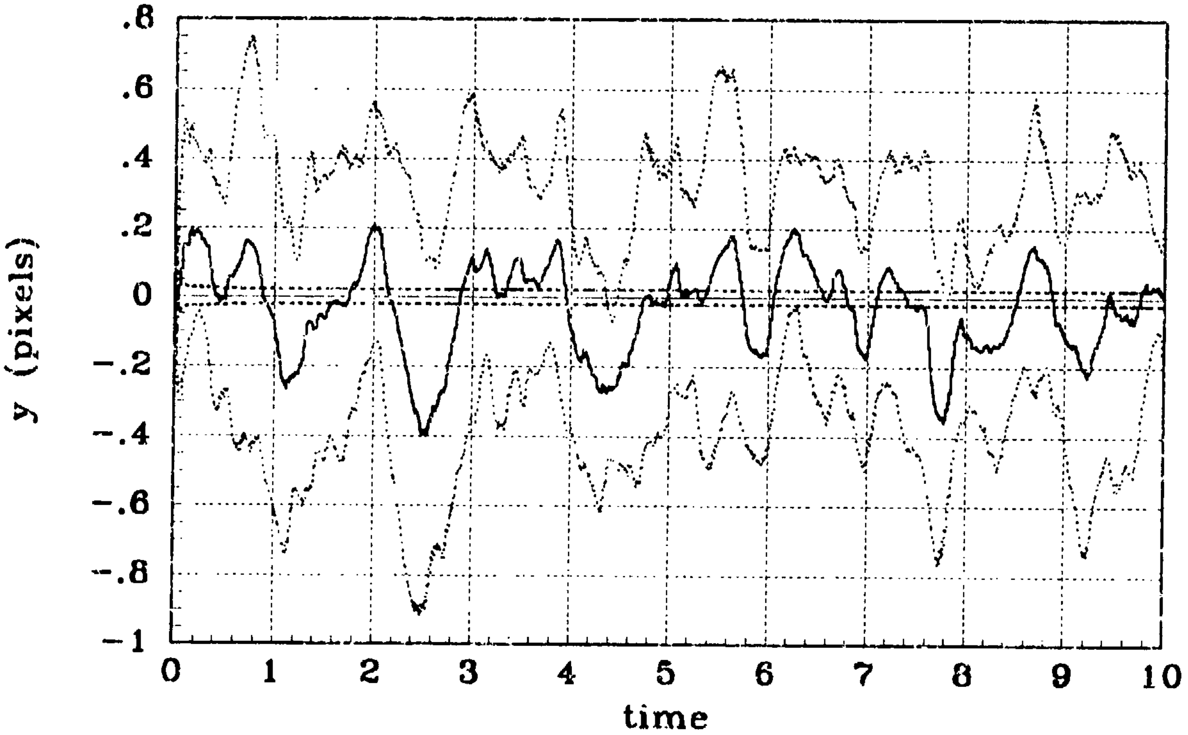


(b) X POSITION STATISTICS – Filter #7, linear

Figure C.16 Simulation #7, θ = 60°, X-Position

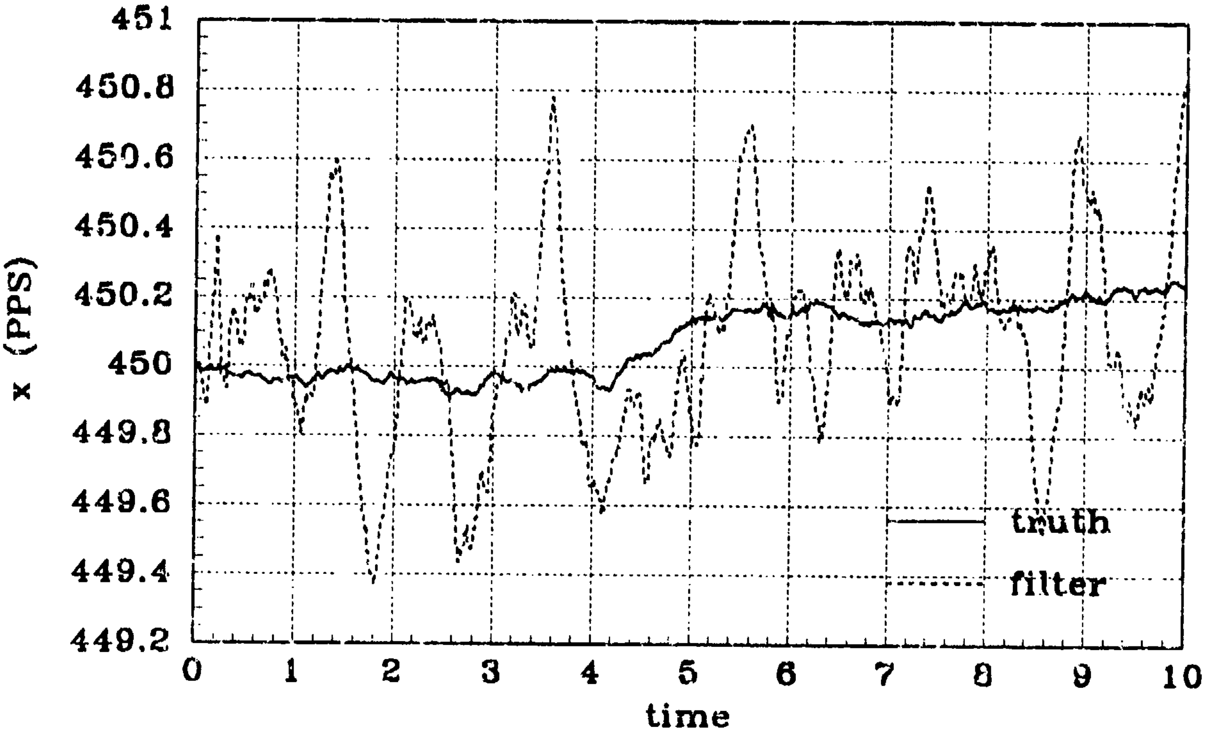


(a) Y POSITION - Filter #7, linear

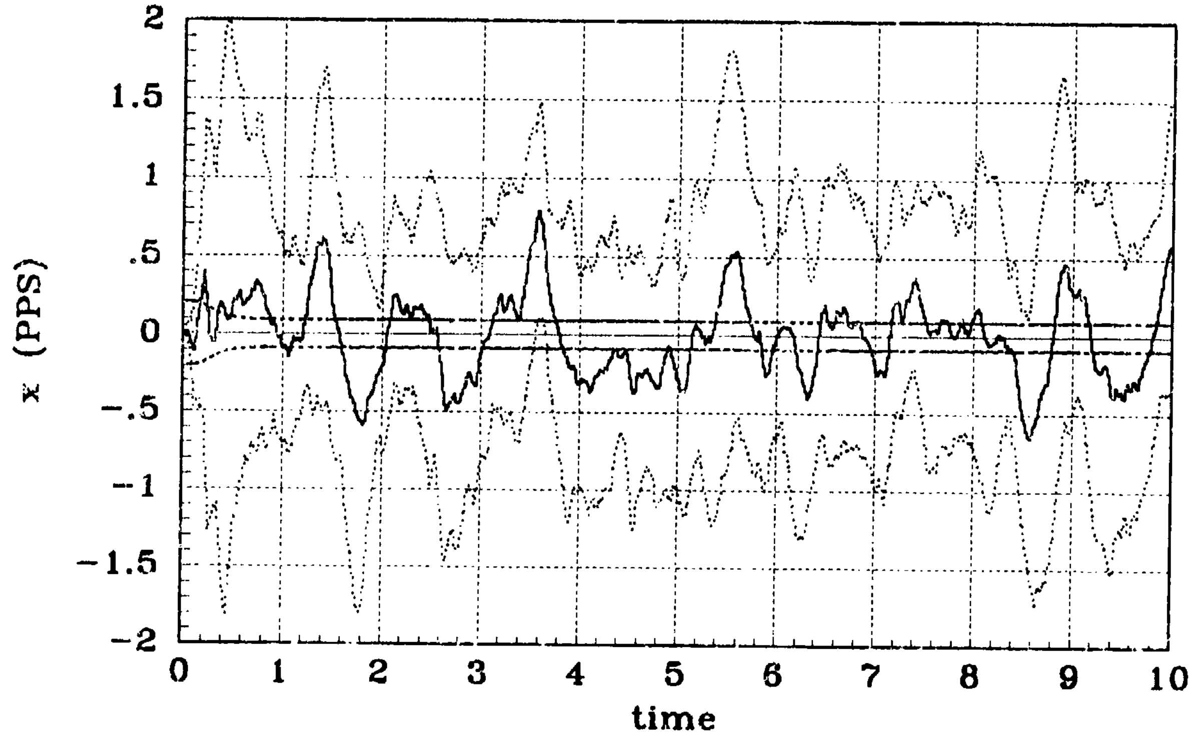


(b) Y POSITION STATISTICS - Filter #7, linear

Figure C.17 Simulation #7, $\theta = 60°$, Y-Position

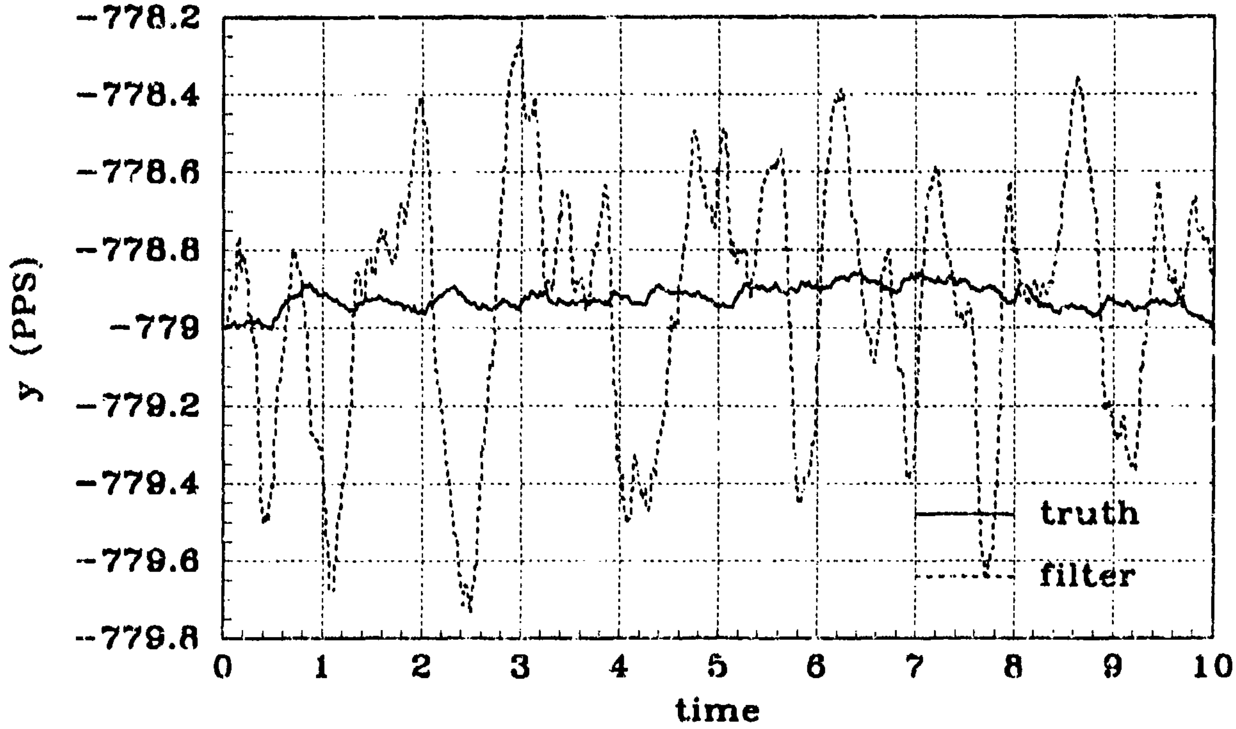


(a) X VELOCITY - Filter #7, linear

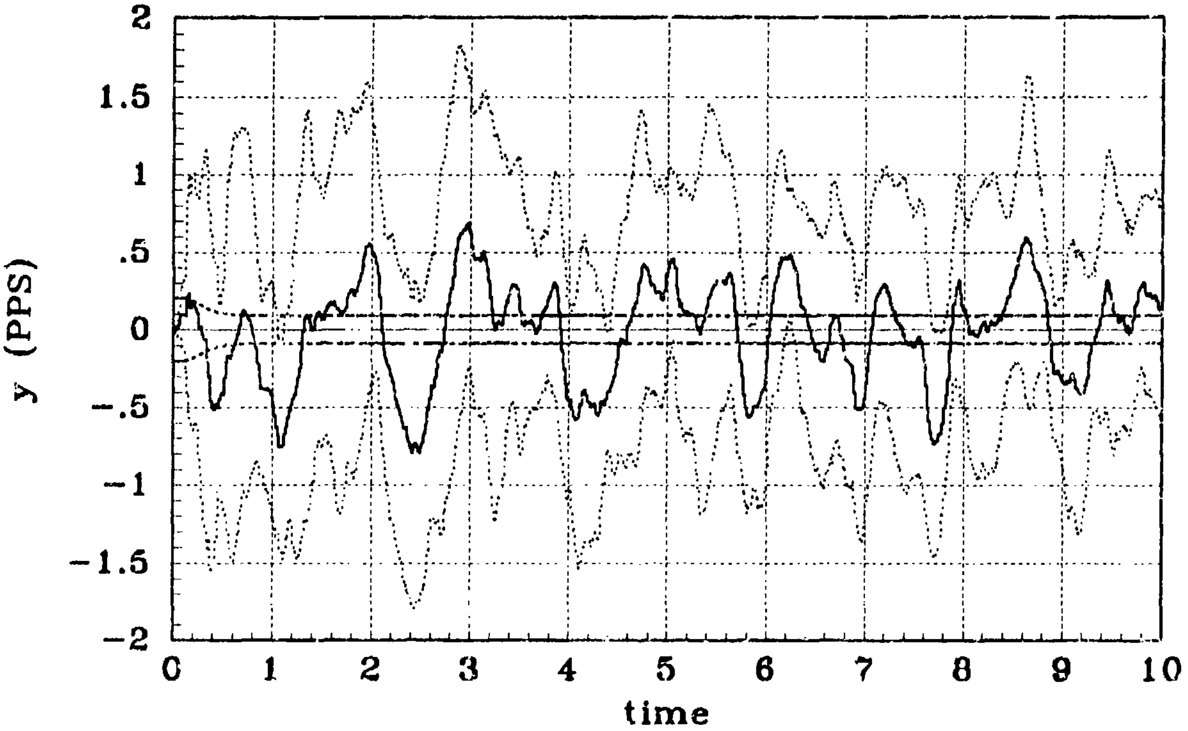


(b) X VELOCITY STATISTICS - Filter #7, linear

Figure C.18 Simulation #7, θ = 60°, X-Velocity

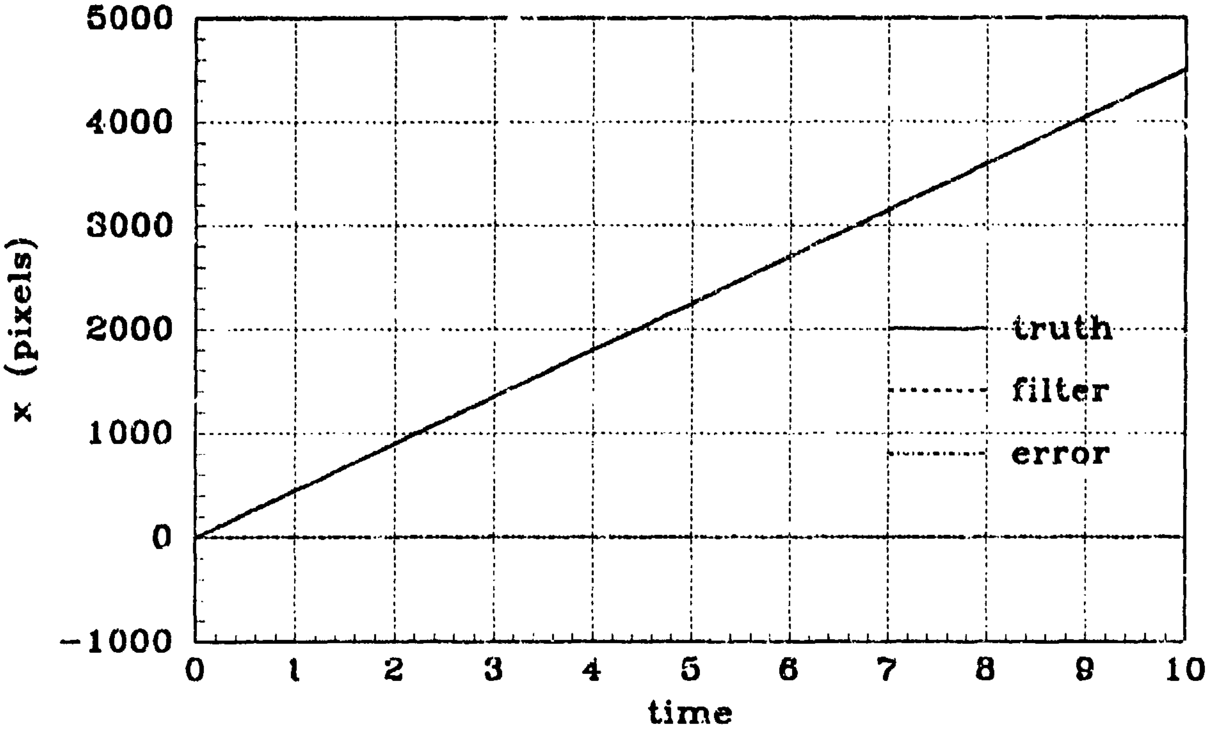


(a) Y VELOCITY – Filter #7, linear

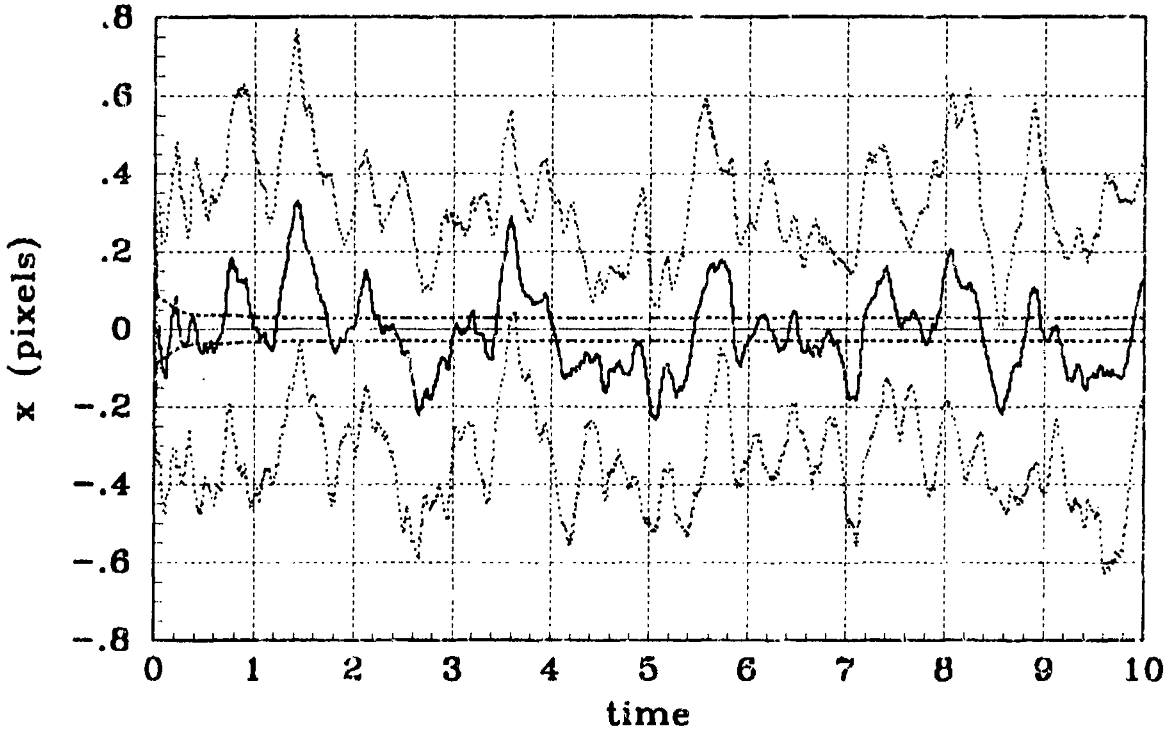


(b) Y VELOCITY STATISTICS – Filter #7, linear

Figure C.19 Simulation #7, θ = 60°, Y-Velocity

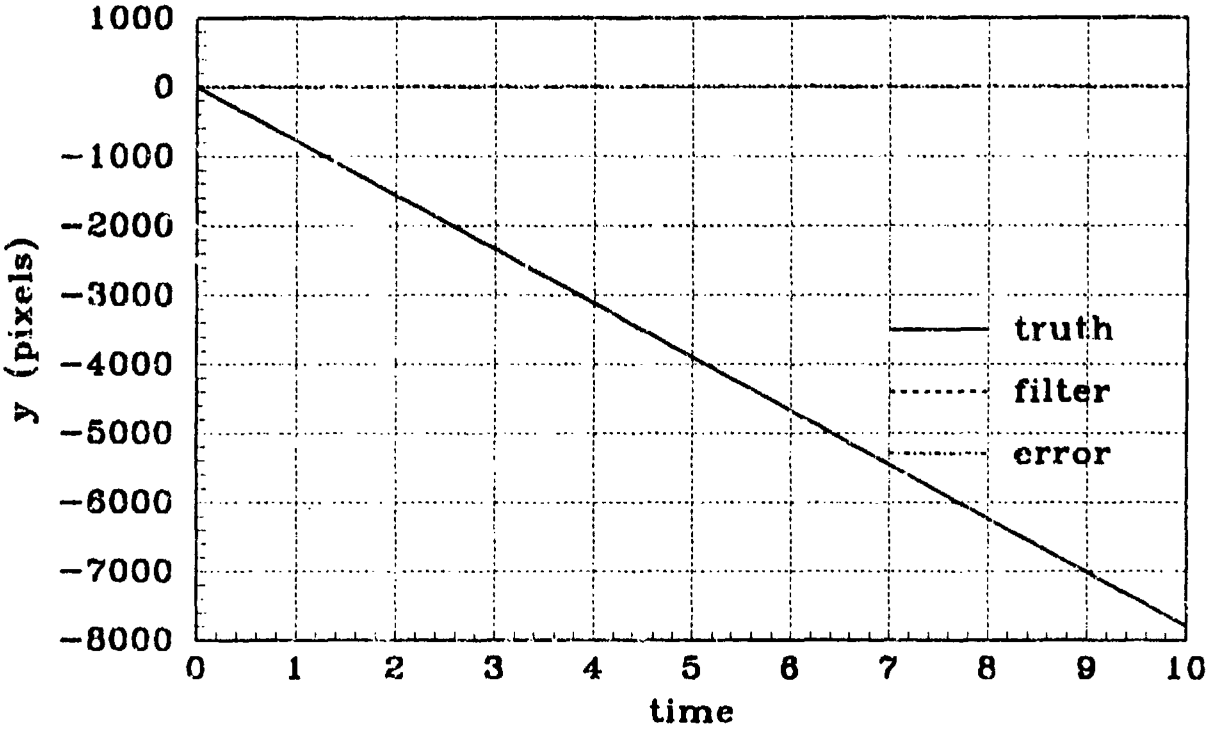


(a) X POSITION – Filter #8, linear, Pogo Cond 4

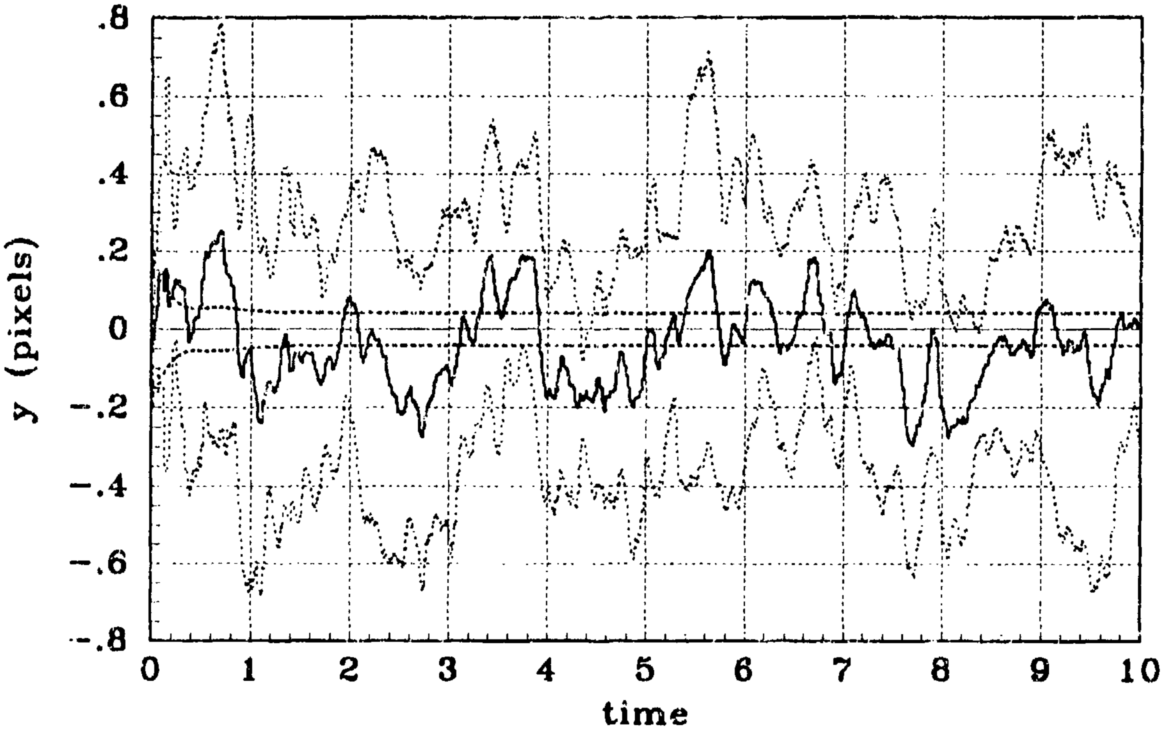


(b) X POSITION STATISTICS – Filter #8, linear, Pogo Cond 4

Figure C.20 Simulation #8, $\theta = 60°$, X-Position

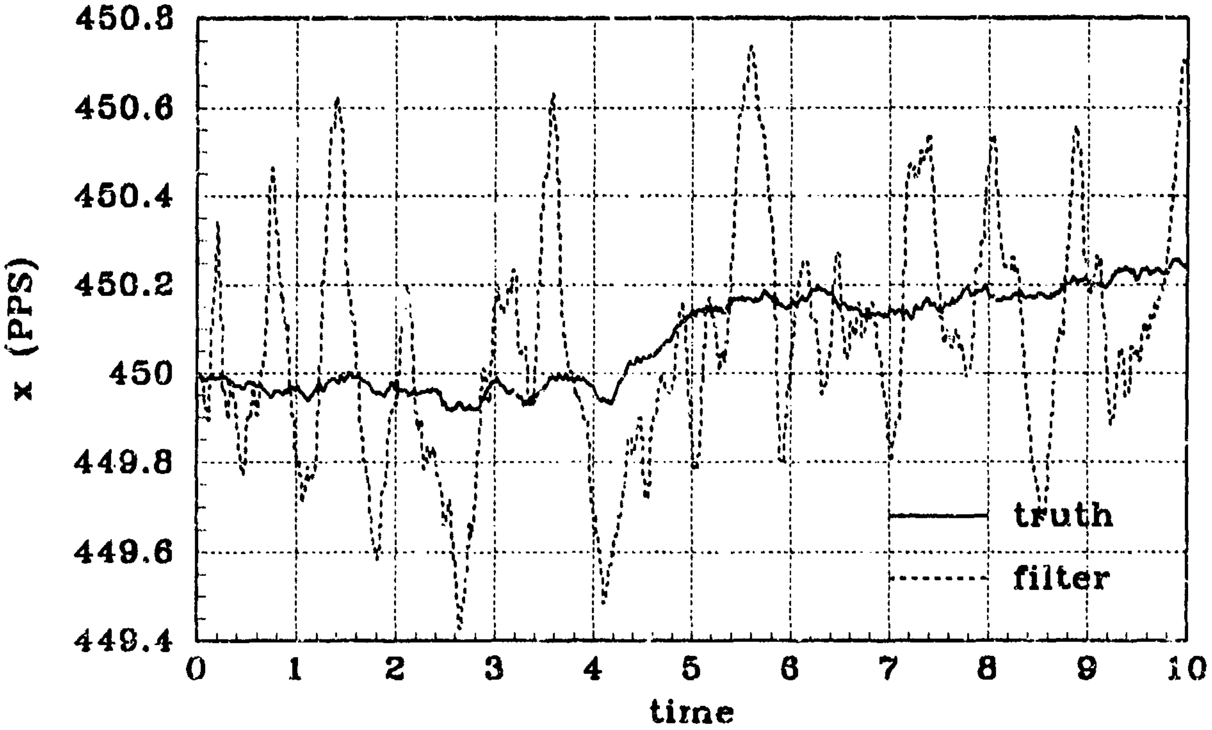


(a) Y POSITION – Filter #8, linear, Pogo Cond 4

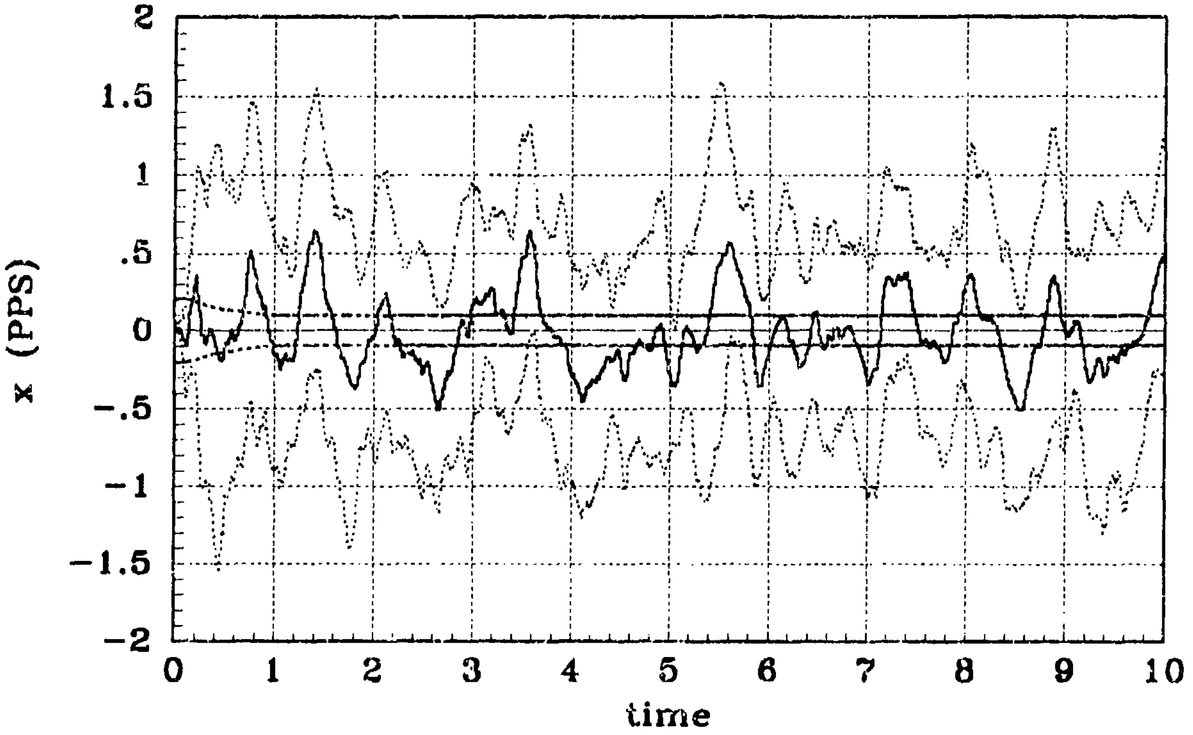


(b) Y POSITION STATISTICS – Filter #8, linear, Pogo Cond 4

Figure C.21 Simulation #8, θ = 60°, Y-Position

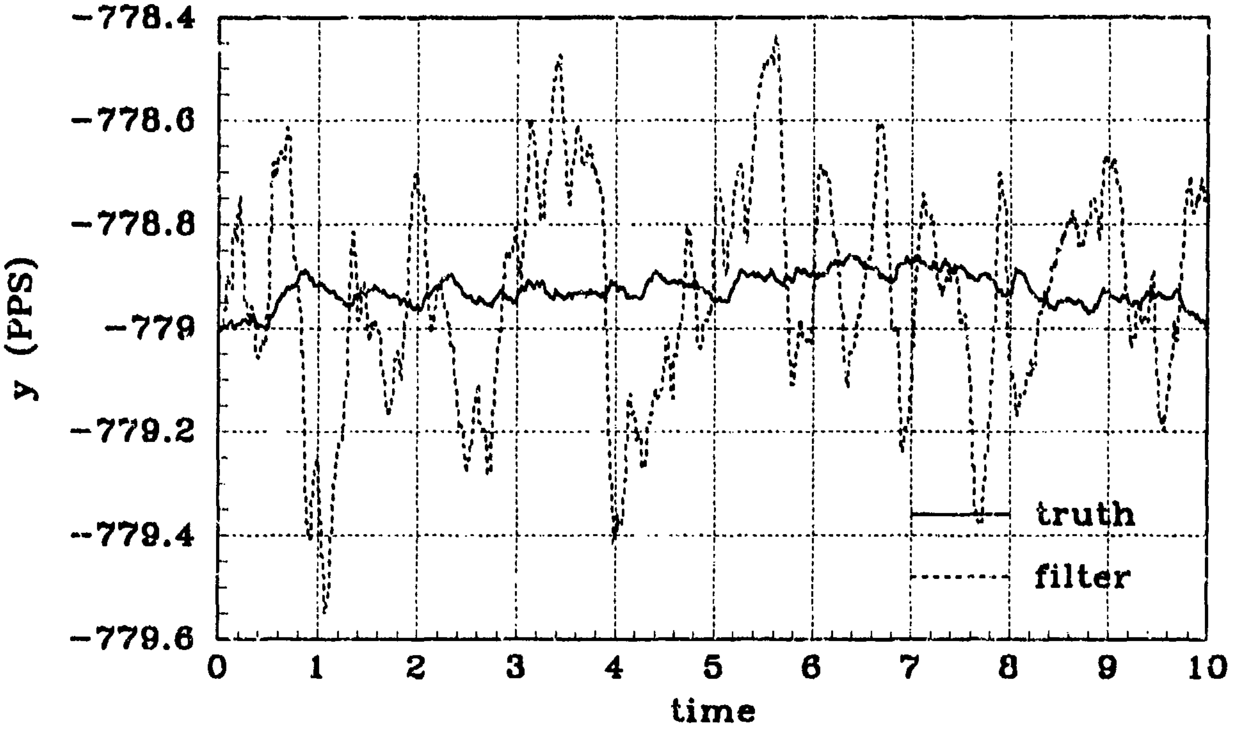


(a) X VELOCITY – Filter #8, linear, Pogo Cond 4

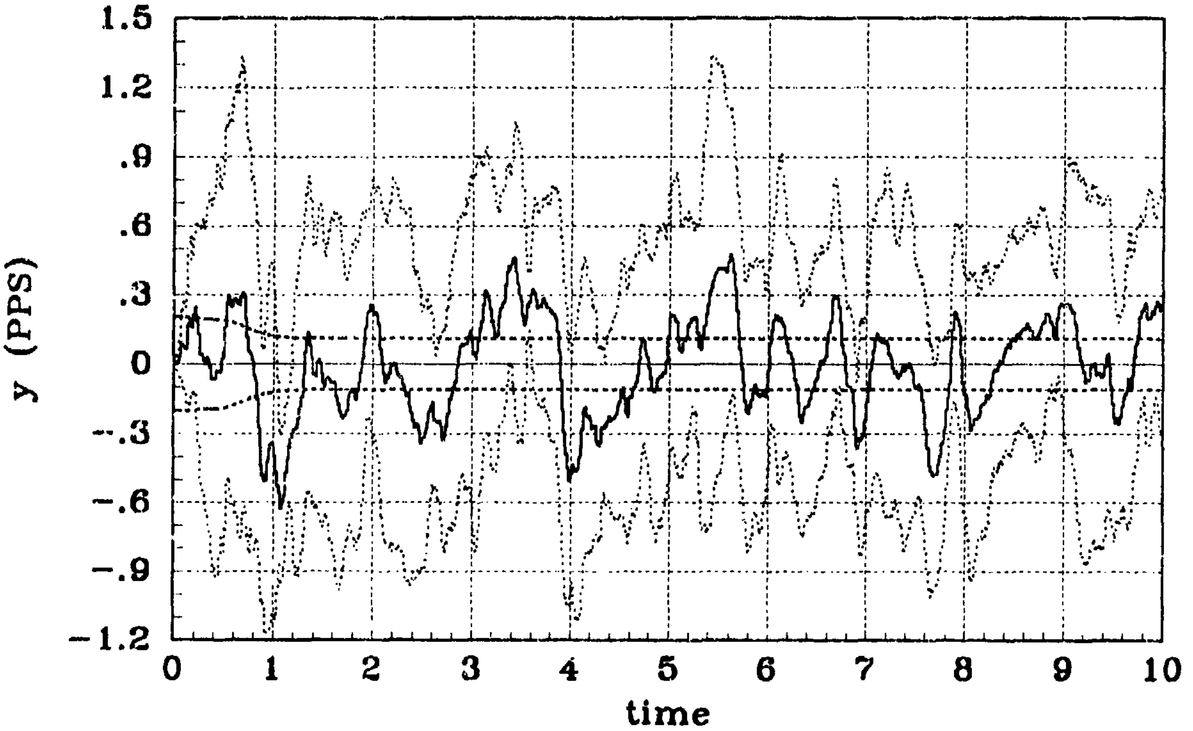


(b) X VELOCITY STATISTICS – Filter #8, linear, Pogo Cond 4

Figure C.22 Simulation #8, θ = 60°, X-Velocity

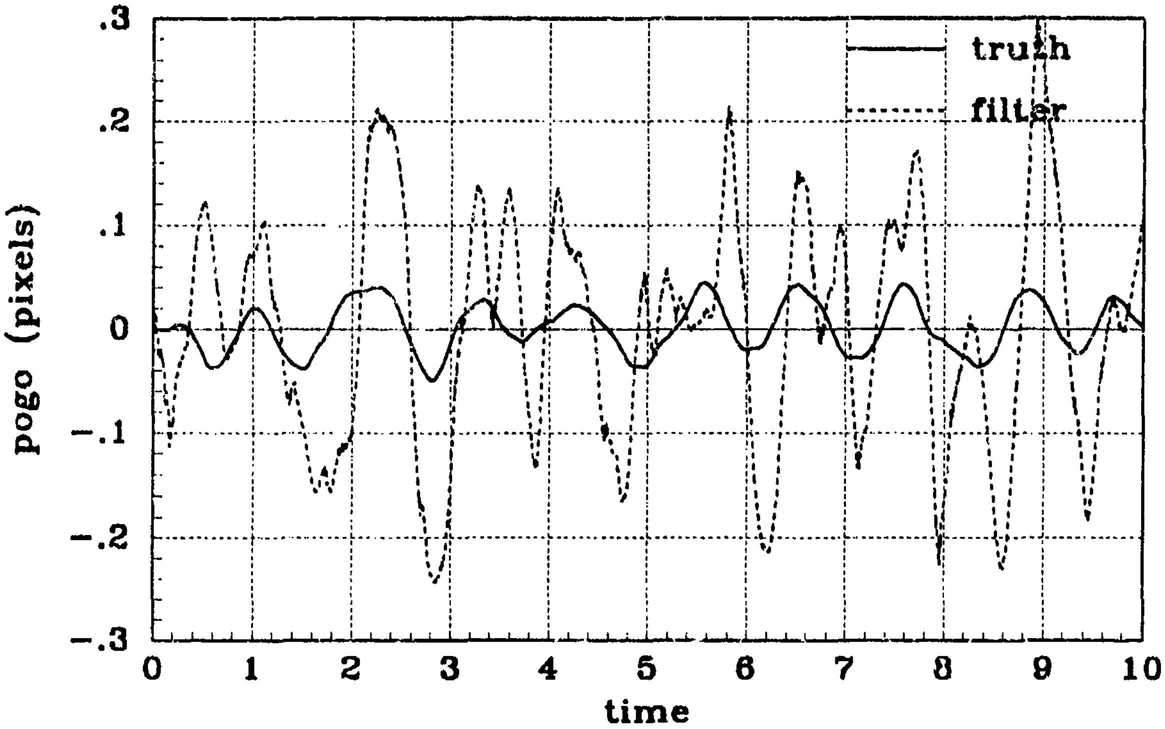


(a) Y VELOCITY – Filter #8, linear, Pogo Cond 4

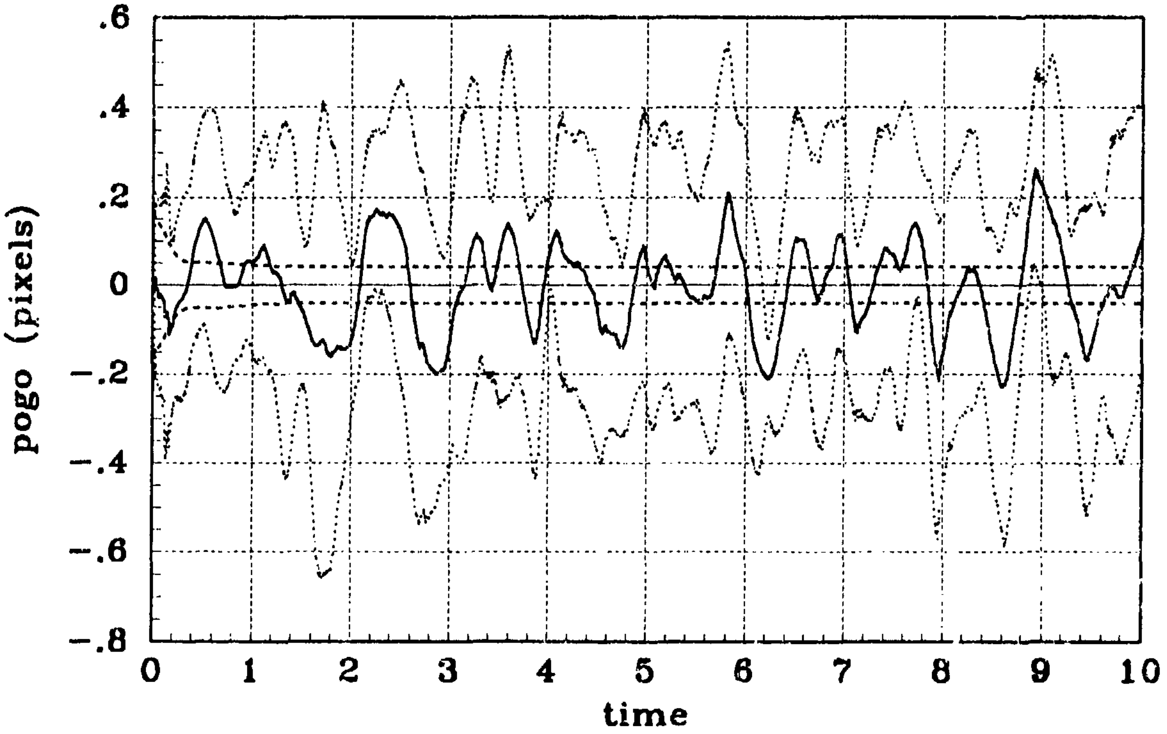


(b) Y VELOCITY STATISTICS – Filter #8, linear, Pogo Cond 4

Figure C.23 Simulation #8, θ = 60°, Y-Velocity

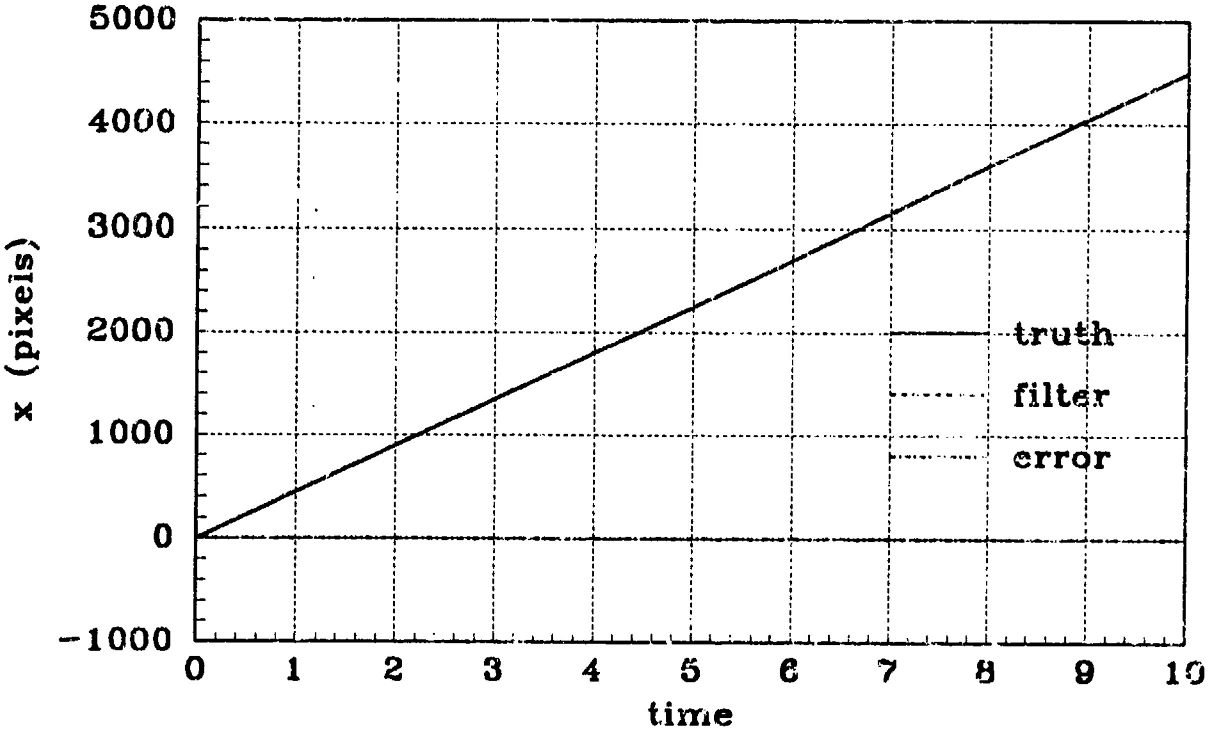


(a) POGO OFFSET - Filter #8, linear, Pogo Cond 4

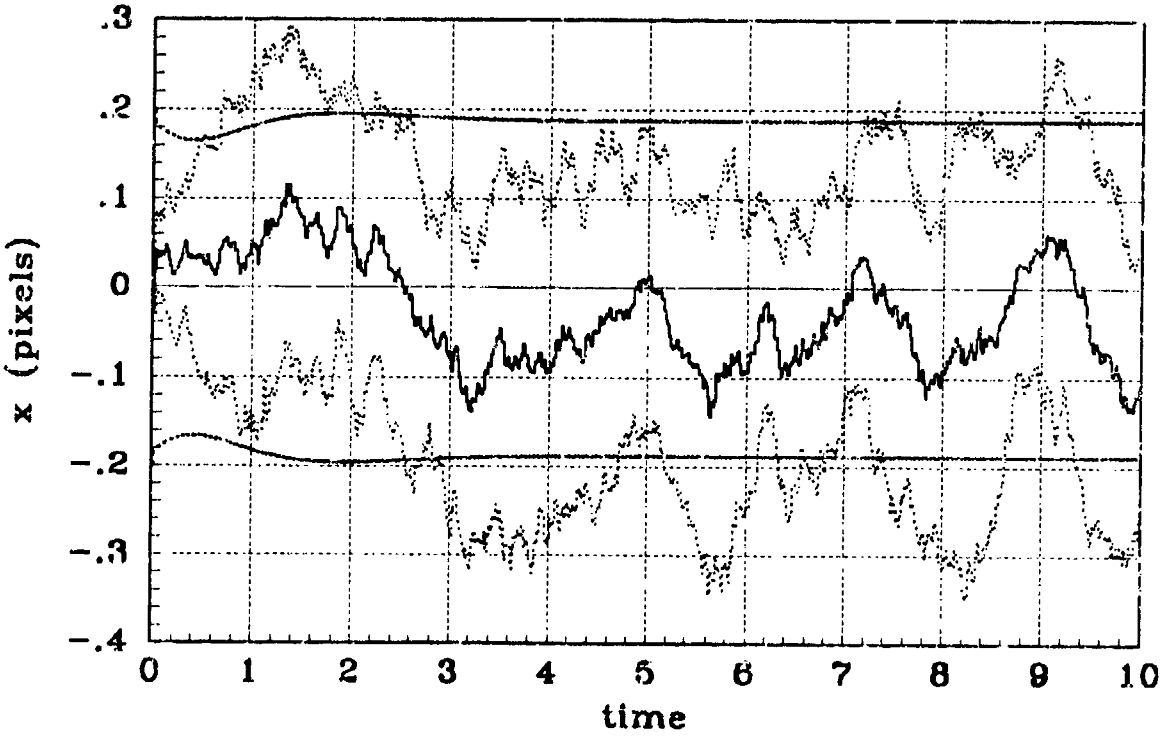


(b) POGO OFFSET STATISTICS - Filter #8, linear, Pogo Cond 4

Figure C.24 Simulation #8, $\theta = 60°$, Pogo Offset

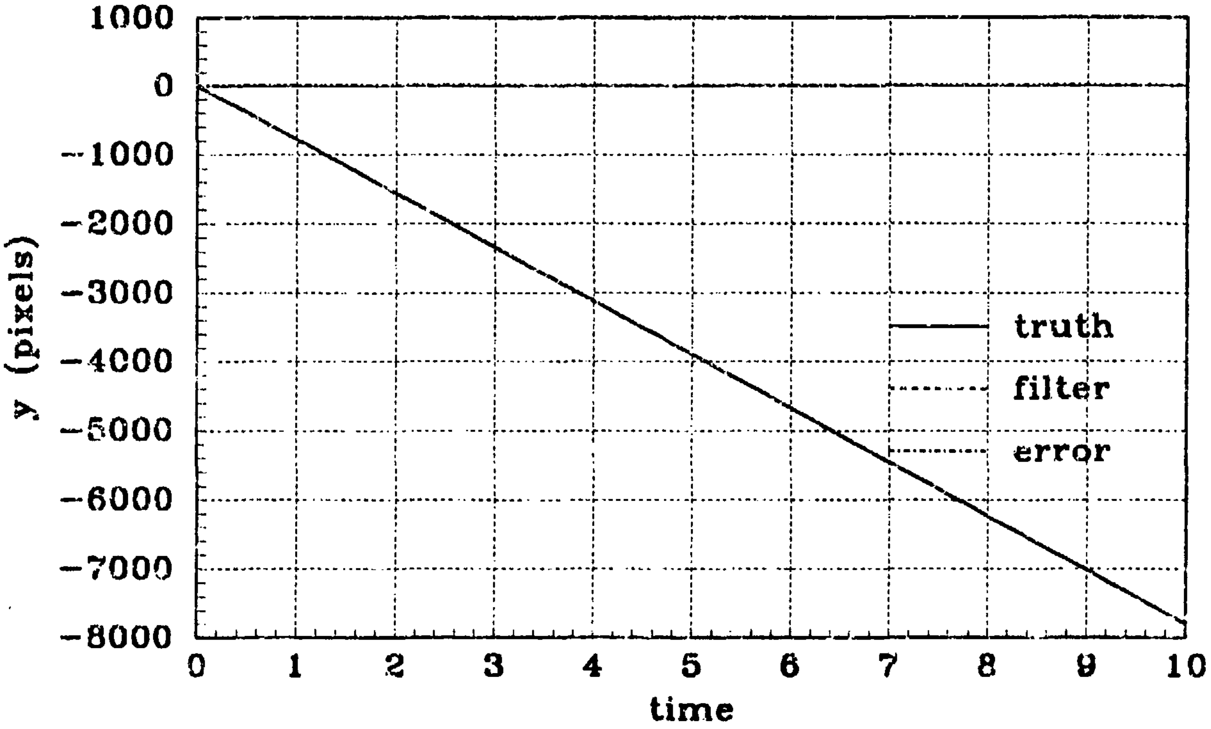


(a) X POSITION – Filter #9, linear

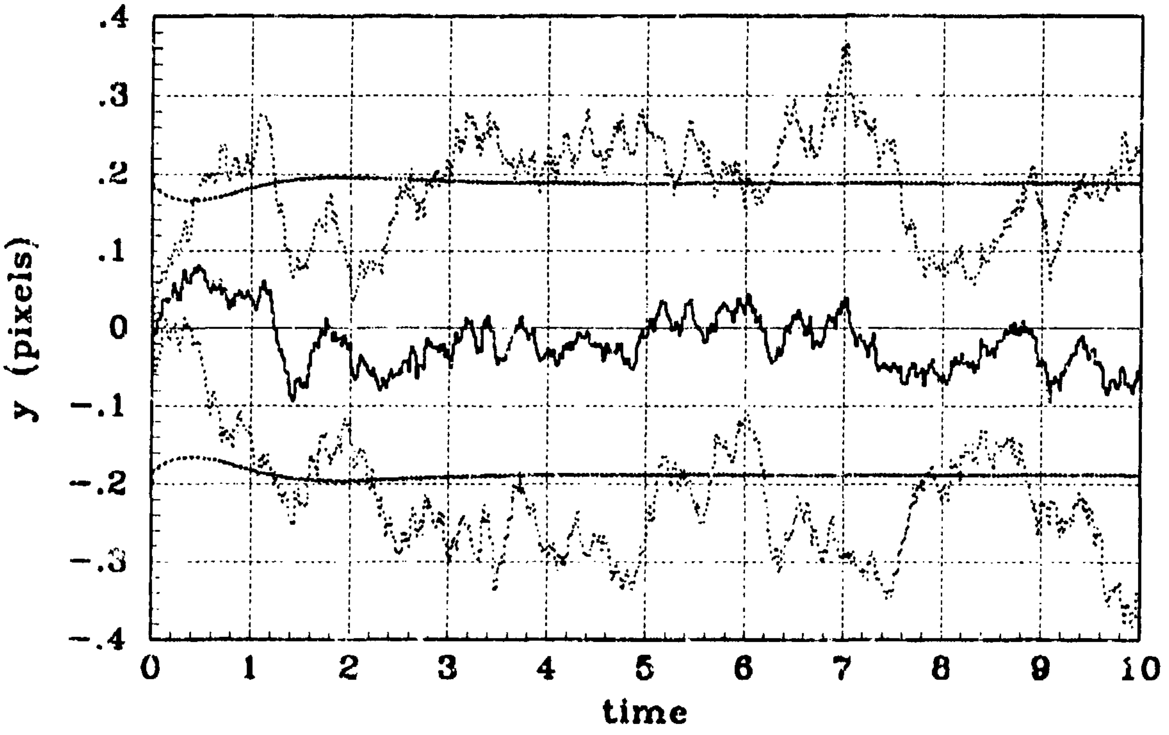


(b) X POSITION STATISTICS – Filter #9, linear

Figure C.25 Simulation #9, θ = 60°, X-Position

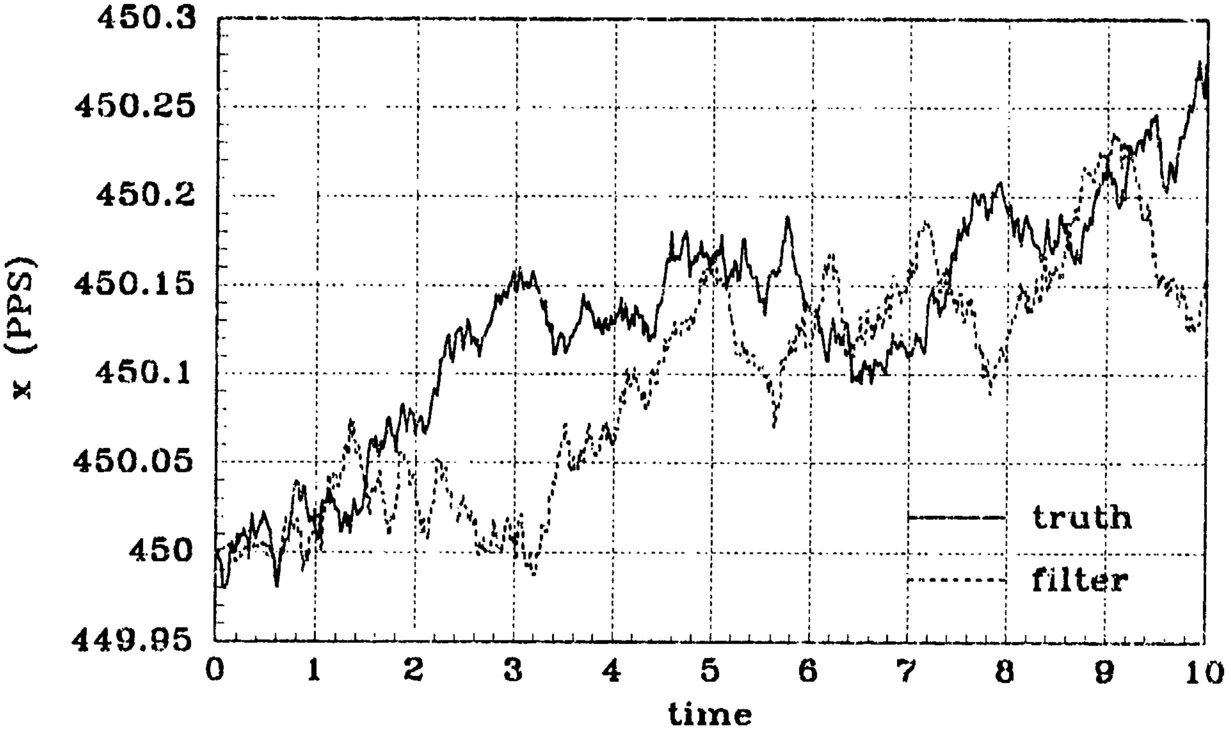


(a) Y POSITION − Filter #9, linear

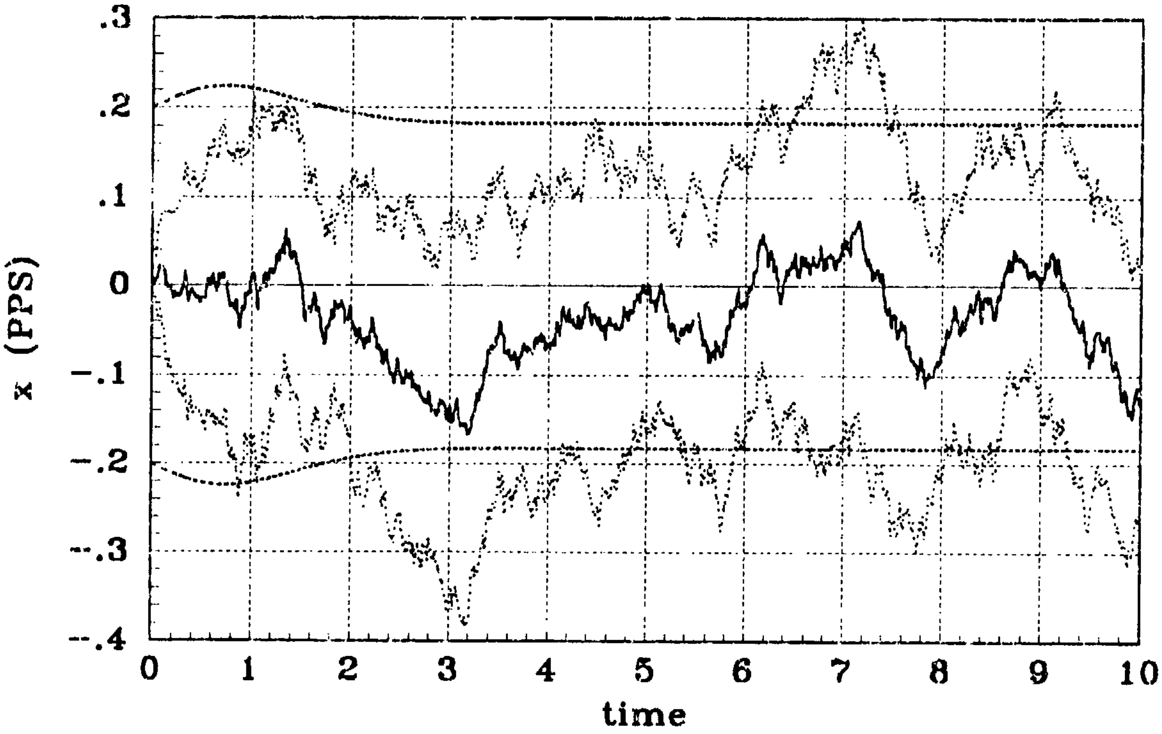


(b) Y POSITION STATISTICS − Filter #9, linear

Figure C.26 Simulation #9, θ = 60°, Y-Position

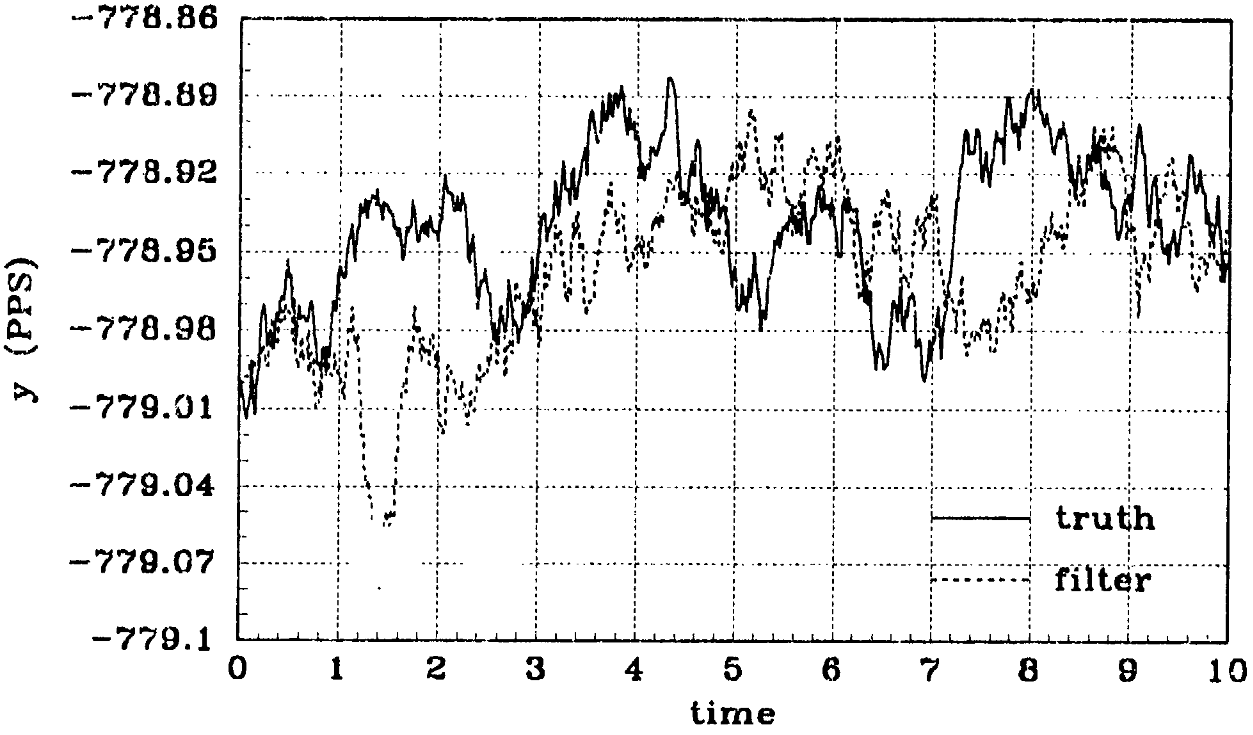


(a) X VELOCITY - Filter #9, linear

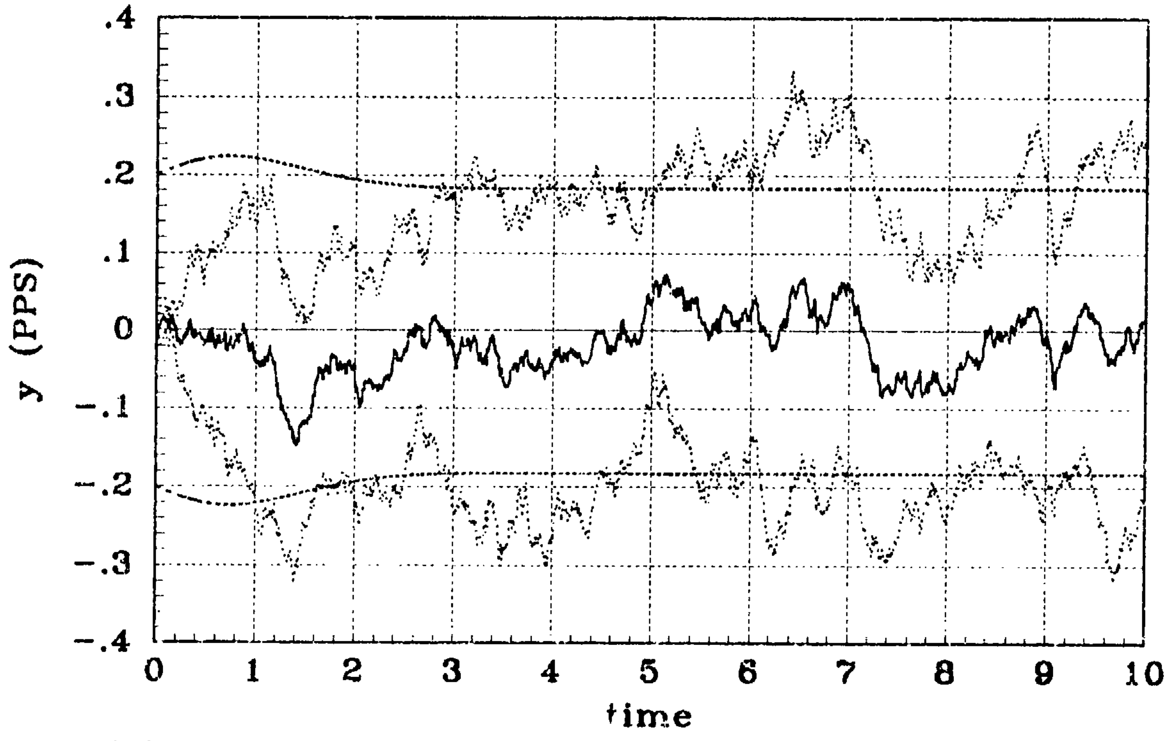


(b) X VELOCITY STATISTICS - Filter #9, linear

Figure C.27 Simulation #9, θ = 60°, X-Velocity

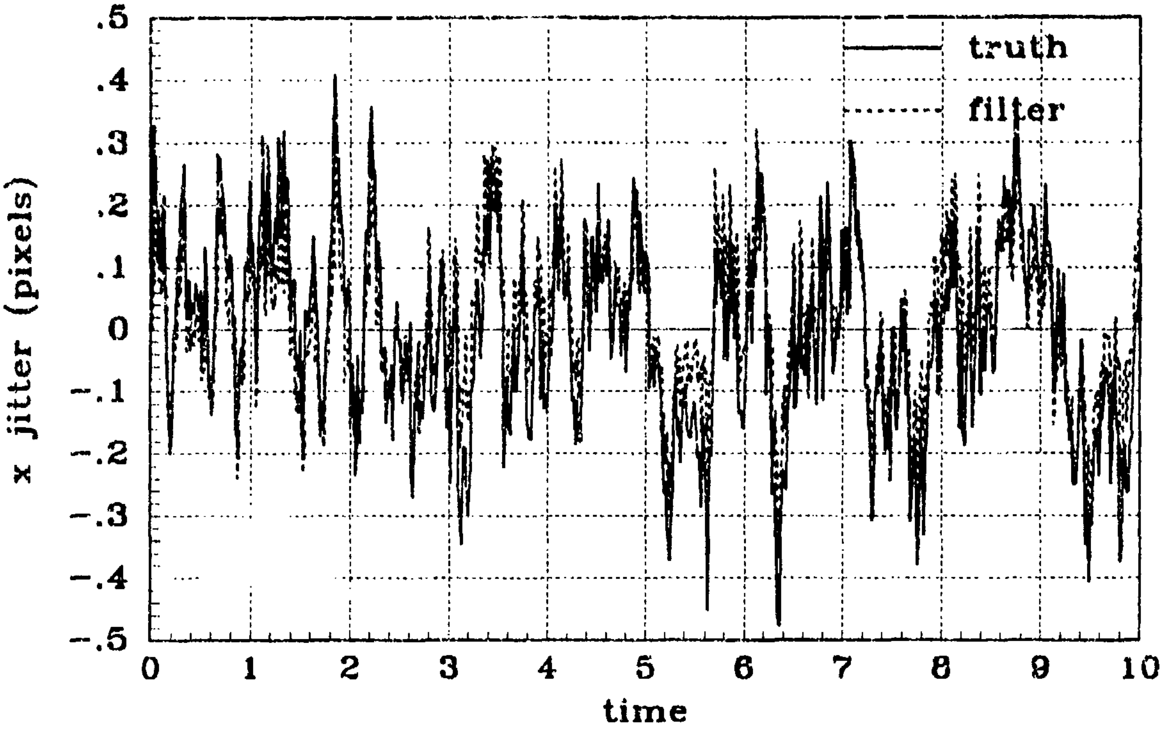


(a) Y VELOCITY − Filter #9, linear

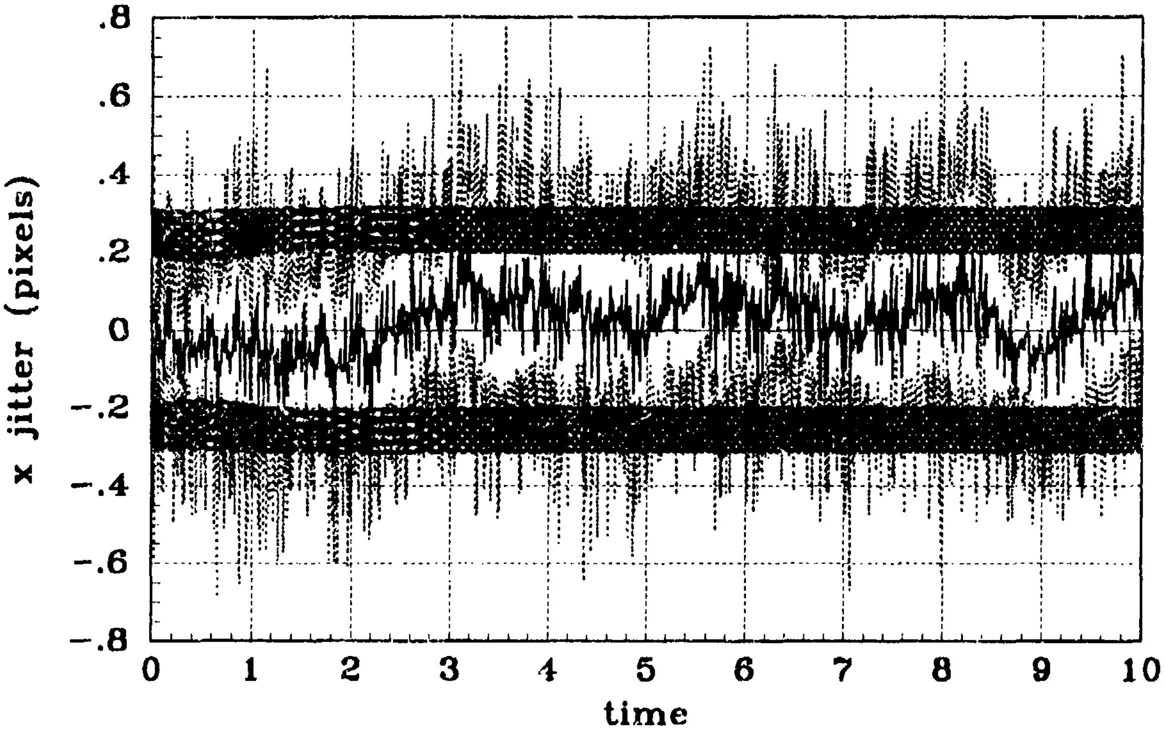


(b) Y VELOCITY STATISTICS − Filter #9, linear

Figure C.28 Simulation #9, θ = 60°, Y-Velocity

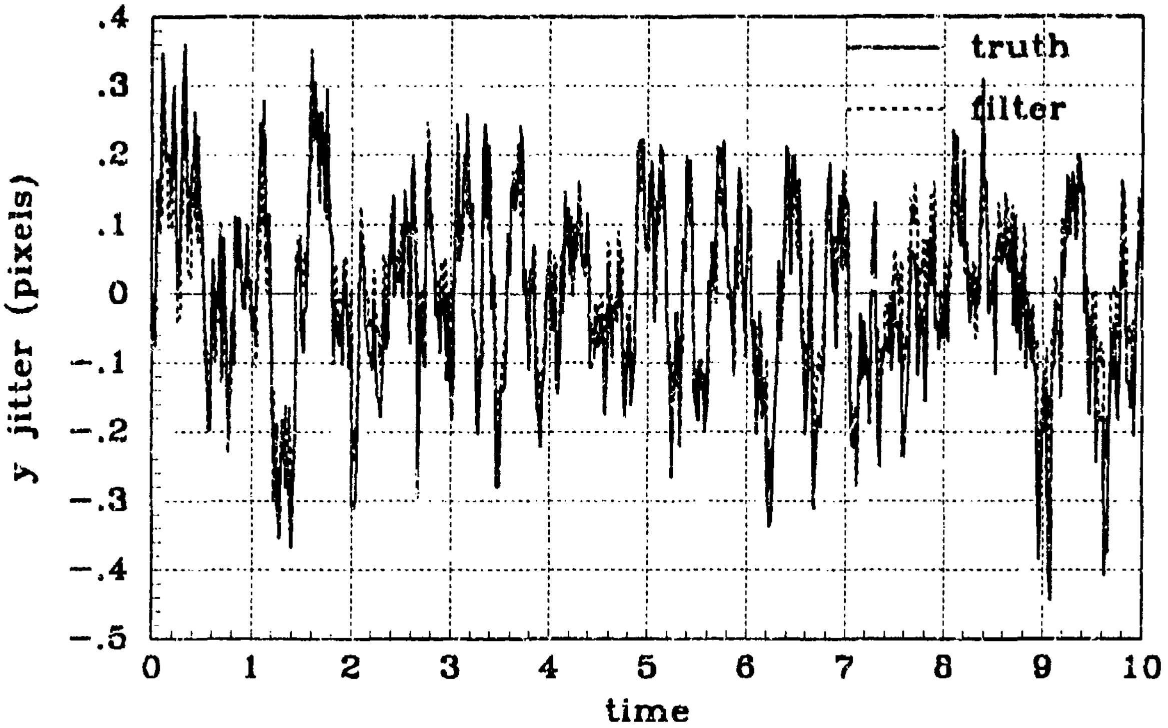


(a) X JITTER - Filter #9, linear

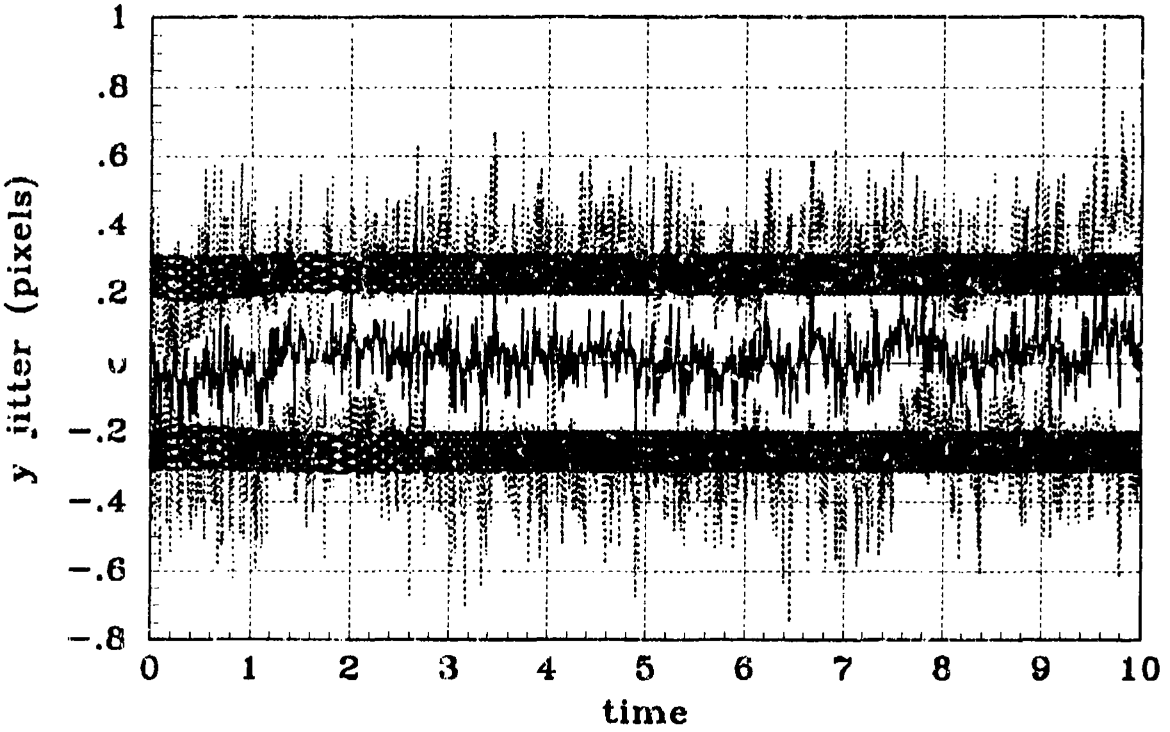


(b) X JITTER STATISTICS - Filter #9, linear

Figure C.29 Simulation #9, θ = 60°, X-Atmospheric Jitter

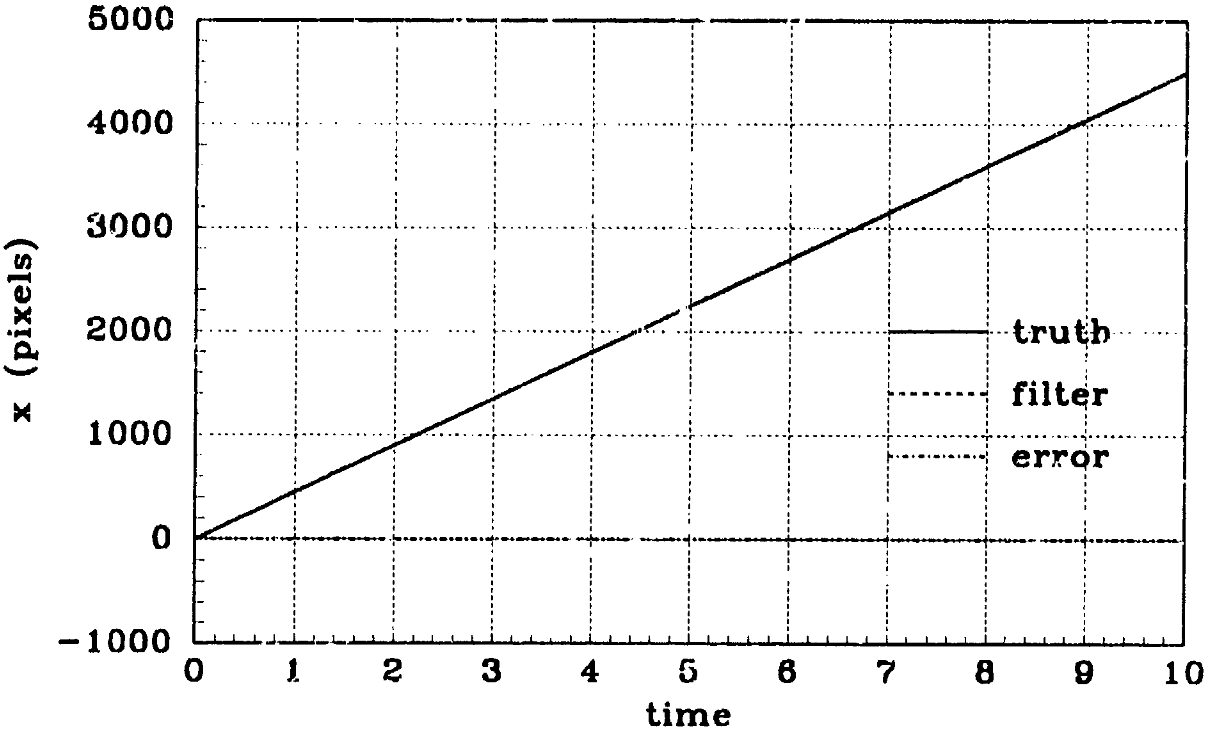


(a) Y JITTER - Filter #9, linear

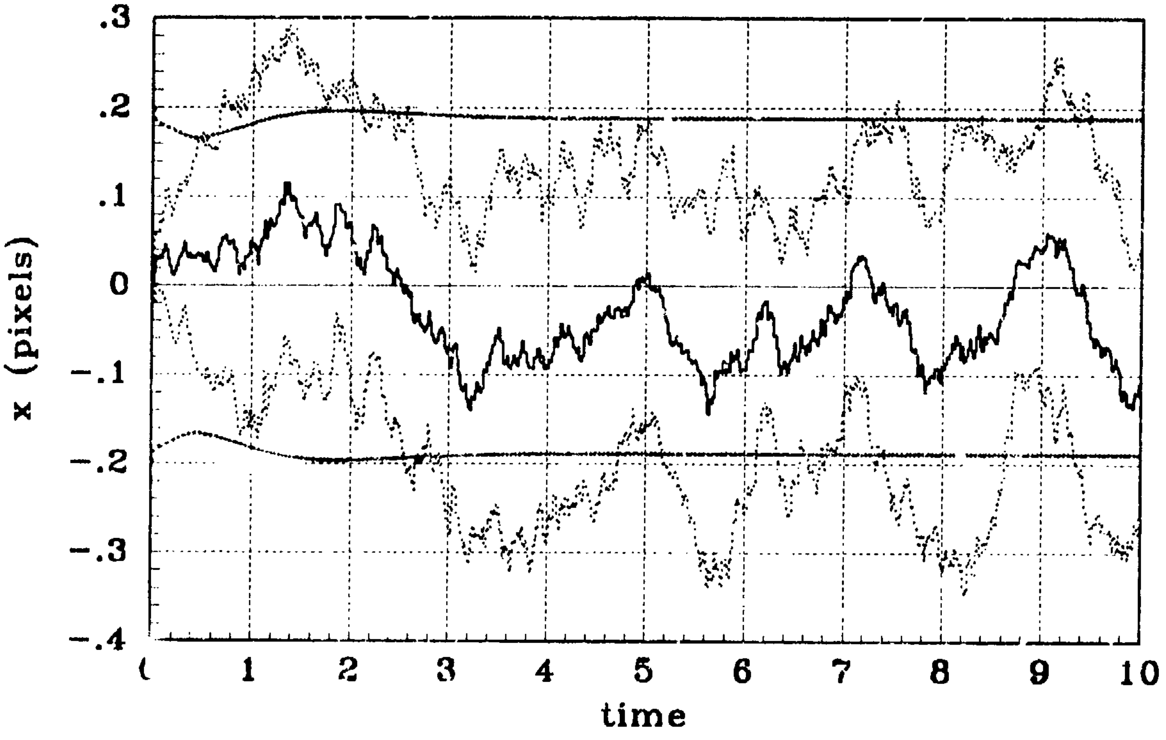


(b) Y JITTER STATISTICS - Filter #9, linear

Figure C.30 Simulation #9, θ = 60°, Y-Atmospheric Jitter

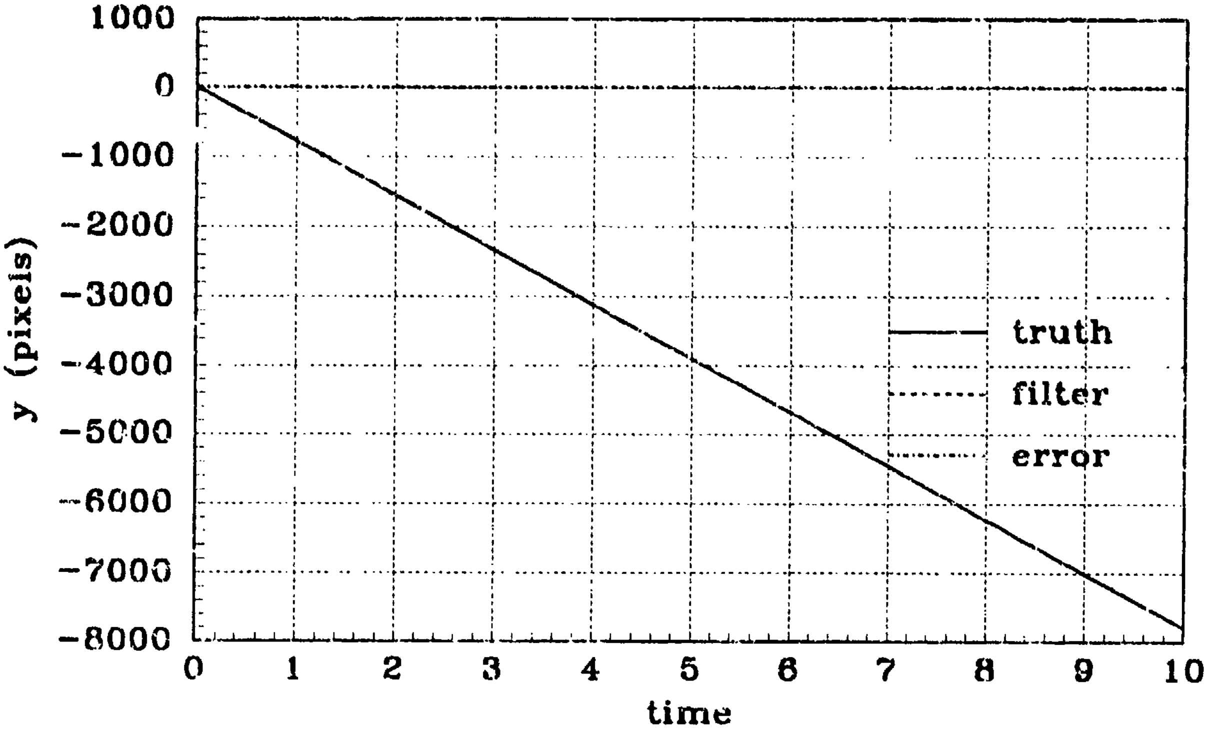


(a) X POSITION – Filter #10, Pogo Cond 4

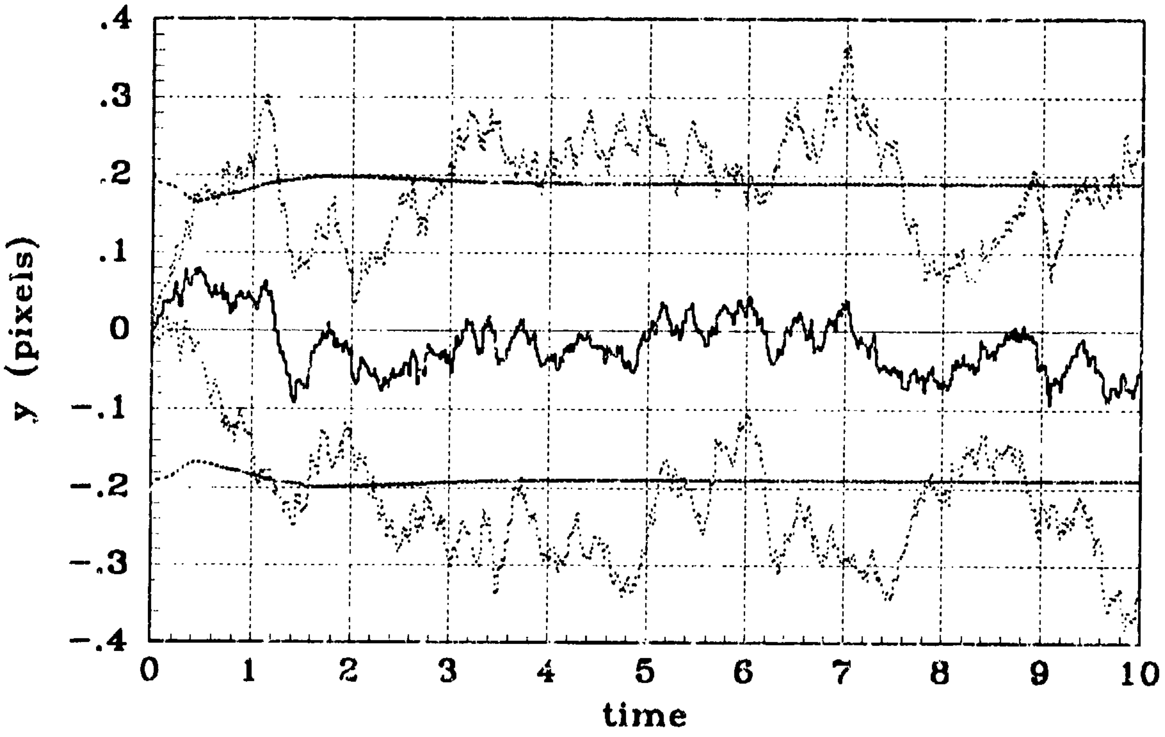


(b) X POSITION STATISTICS – Filter #10, Pogo Cond 4

Figure C.31 Simulation #10, θ = 60°, X-Position

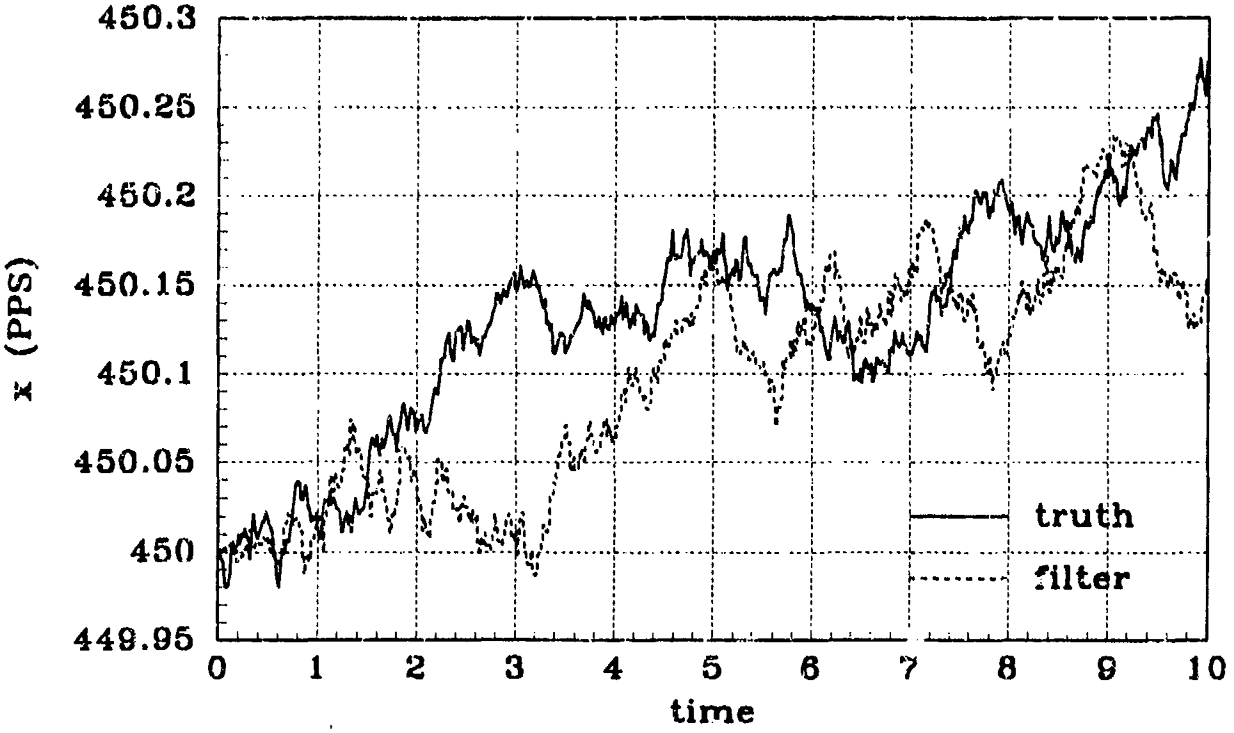


(a) Y POSITION - Filter #10, Pogo Cond 4

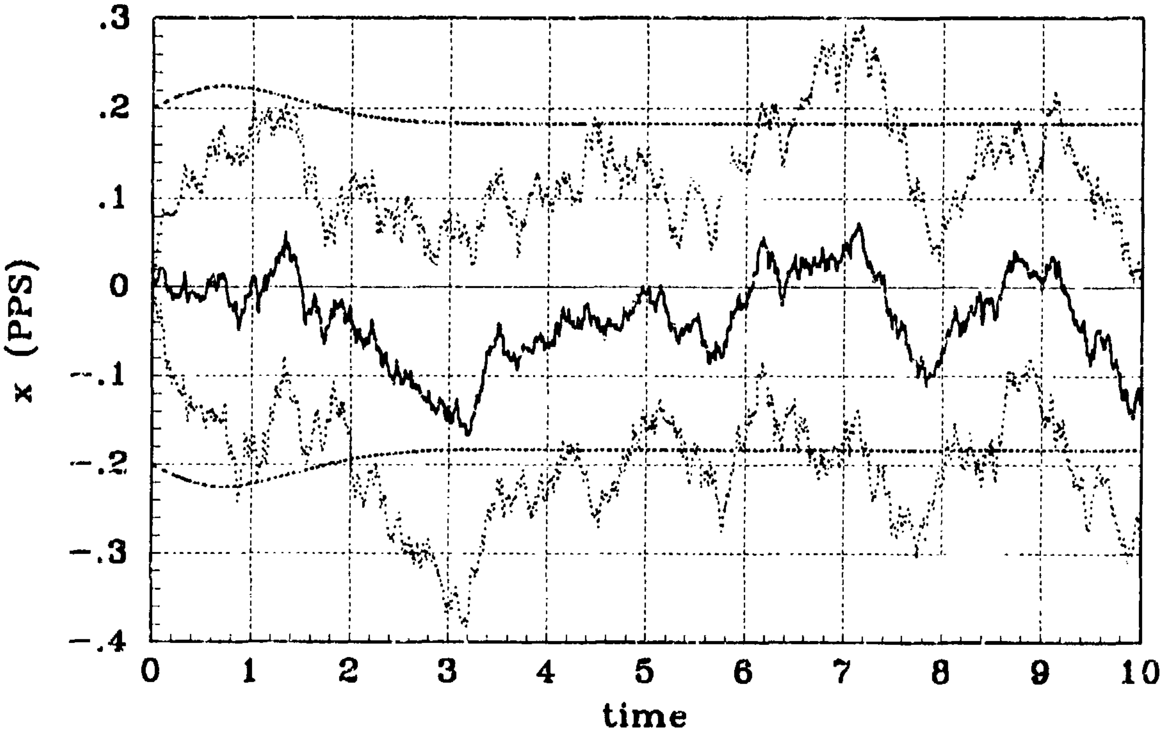


(b) Y POSITION STATISTICS - Filter #10, Pogo Cond 4

Figure C.32 Simulation #10, θ = 60°, Y-Position

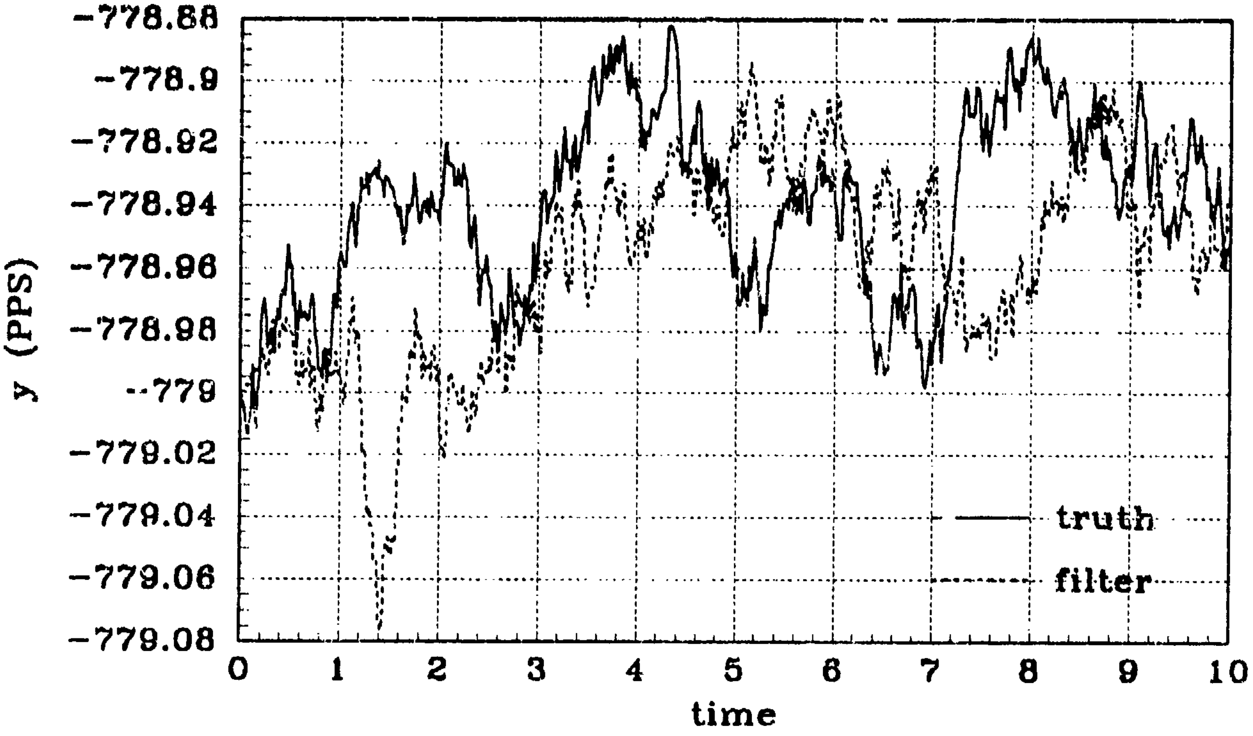


(a) X VELOCITY - Filter #10, Pogo Cond 4

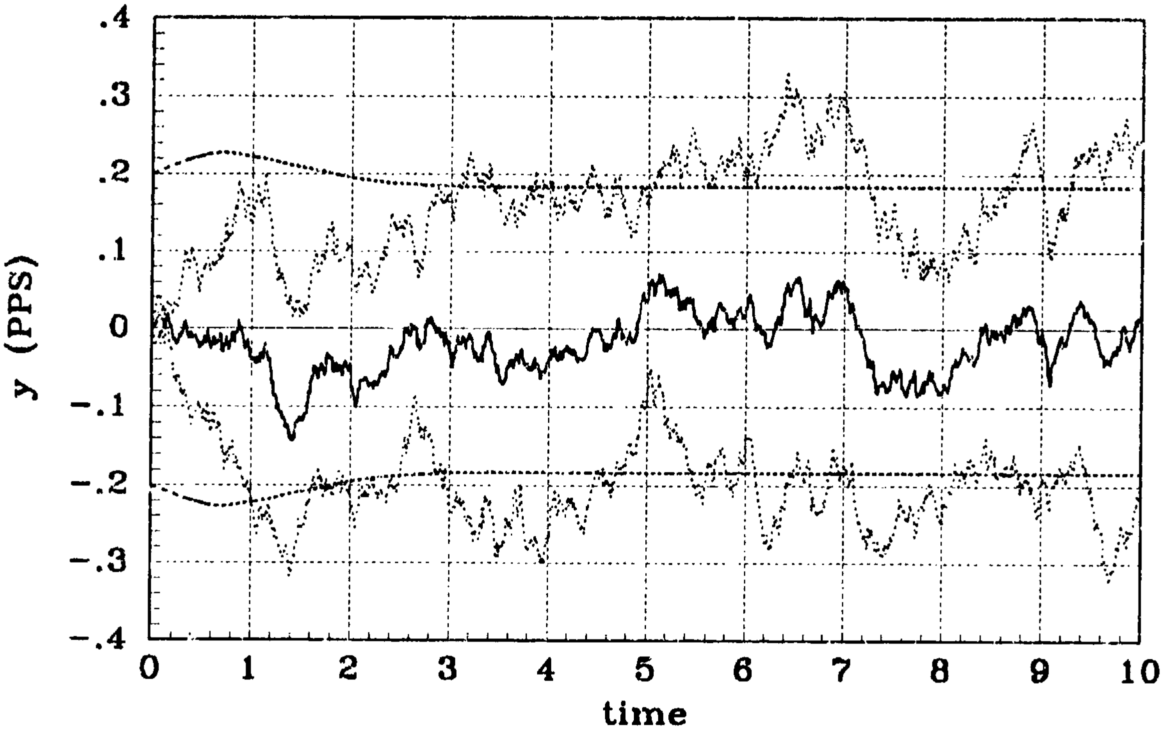


(b) X VELOCITY STATISTICS - Filter #10, Pogo Cond 4

Figure C.33 Simulation #10, θ = 60°, X-Velocity

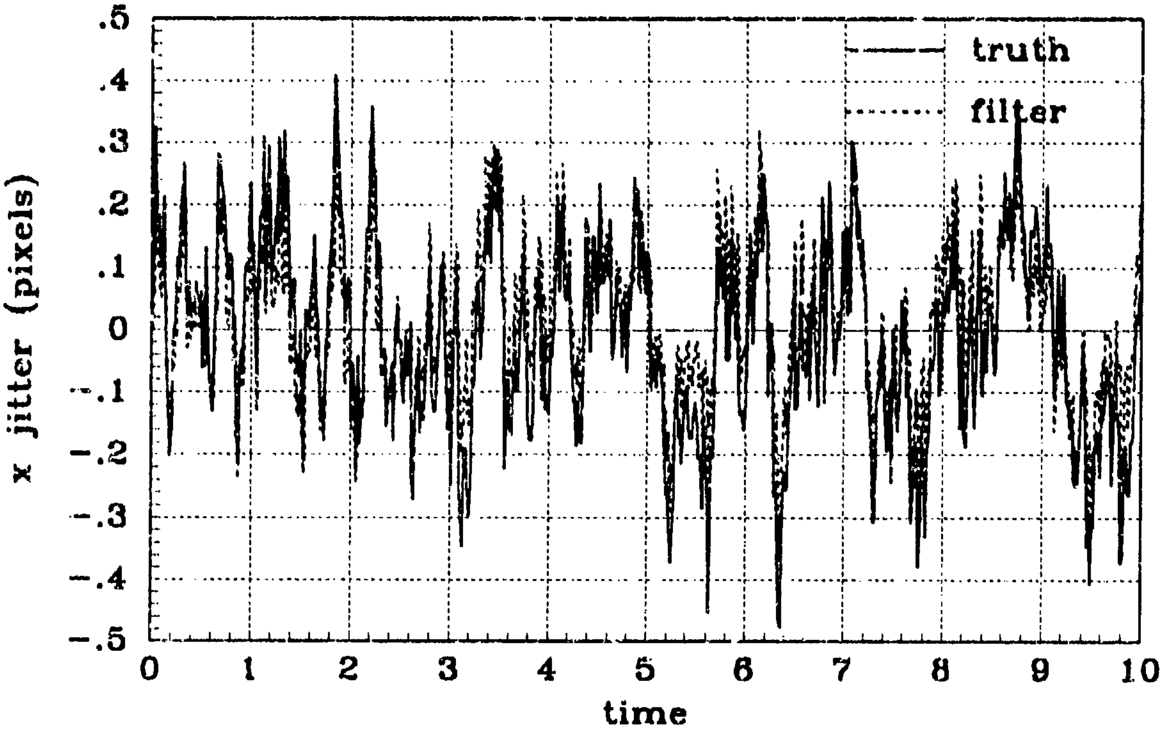


(a) Y VELOCITY – Filter #10, Pogo Cond 4

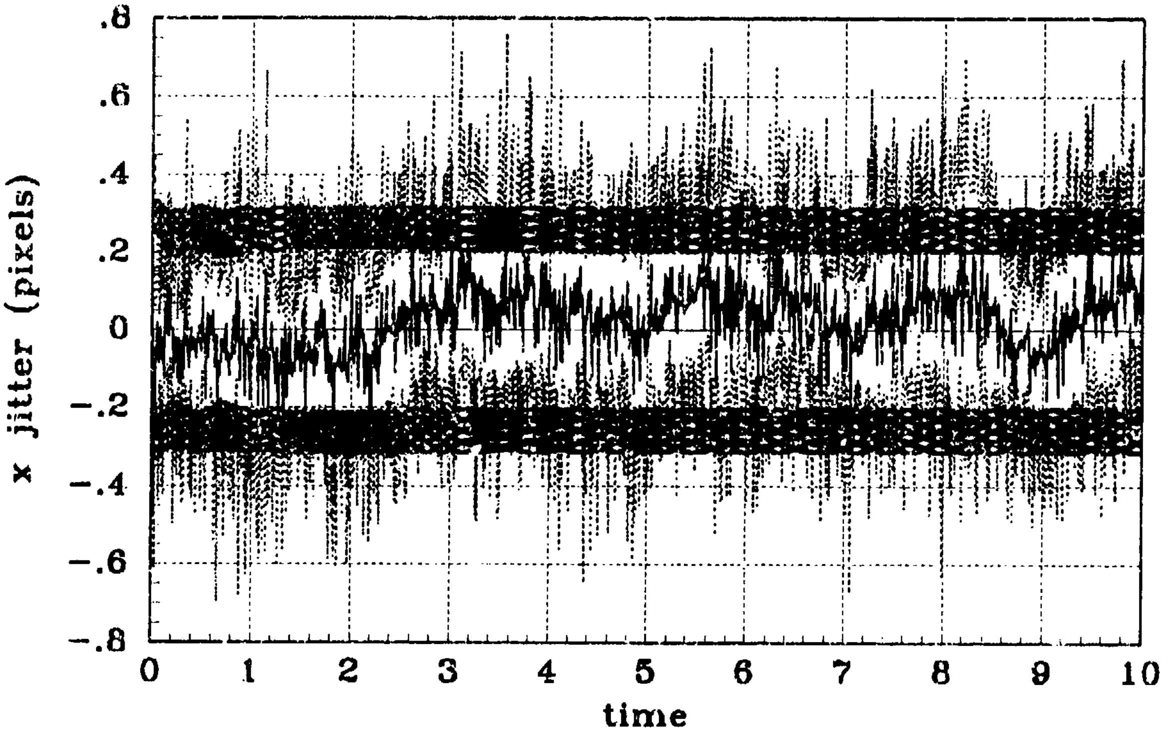


(b) Y VELOCITY STATISTICS – Filter #10, Pogo Cond 4

Figure C.34 Simulation #10, θ = 60°, Y-Velocity

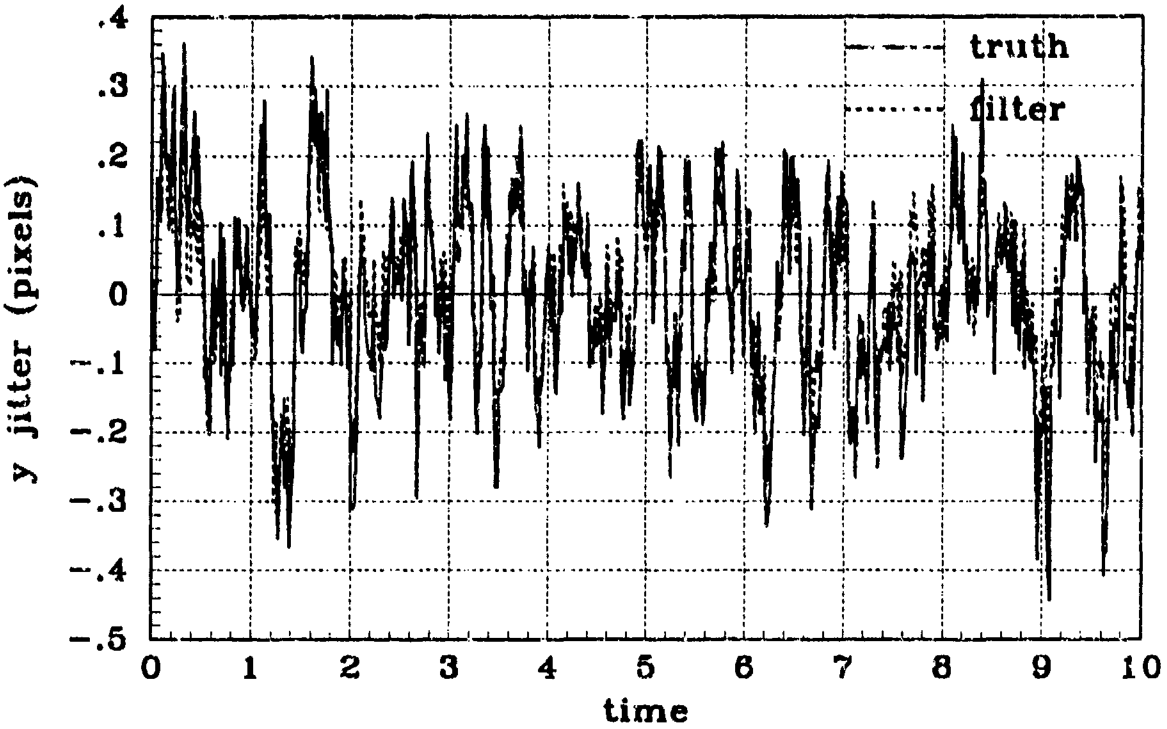


(a) X JITTER – Filter #10, Pogo Cond 4

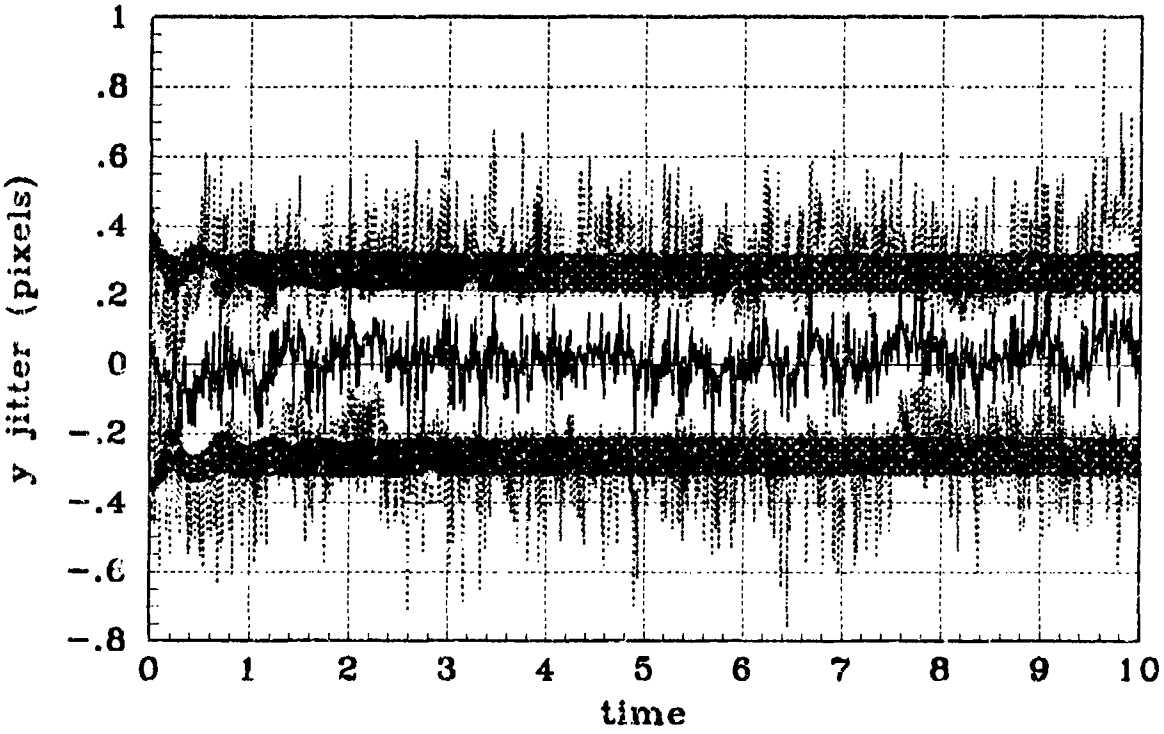


(b) X JITTER STATISTICS – Filter #10, Pogo Cond 4

Figure C.35 Simulation #10, θ = 60°, X-Atmospheric Jitter

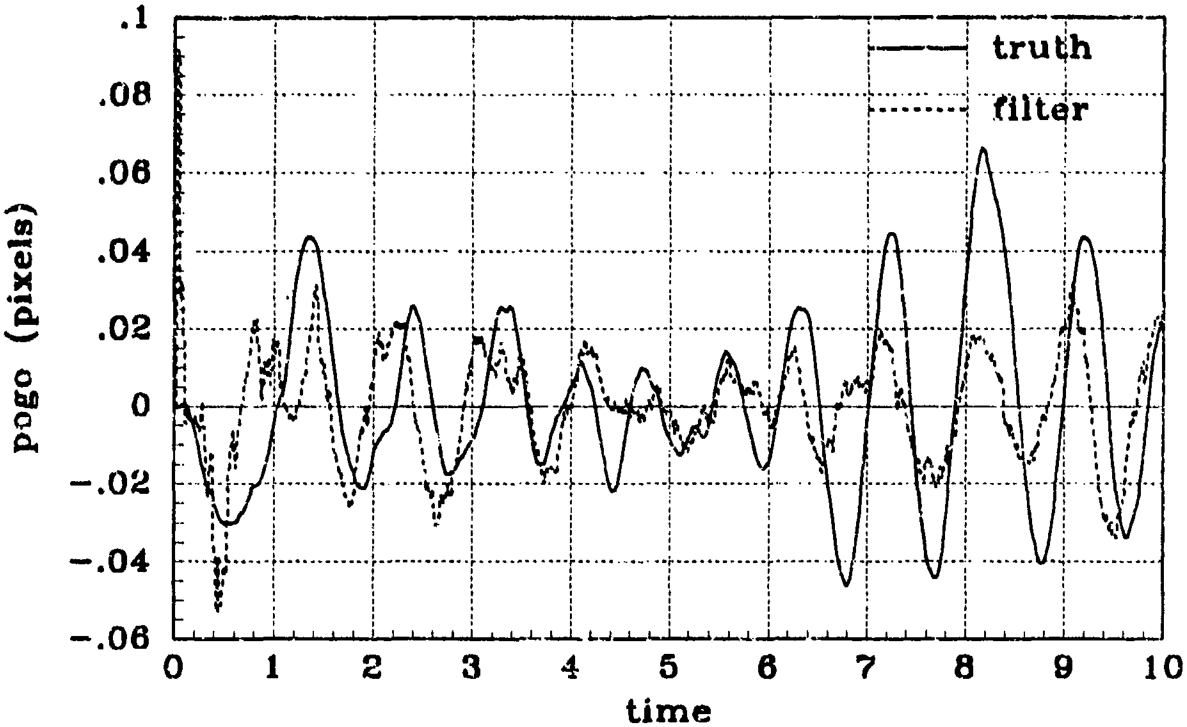


(a) Y JITTER - Filter #10, Pogo Cond 4

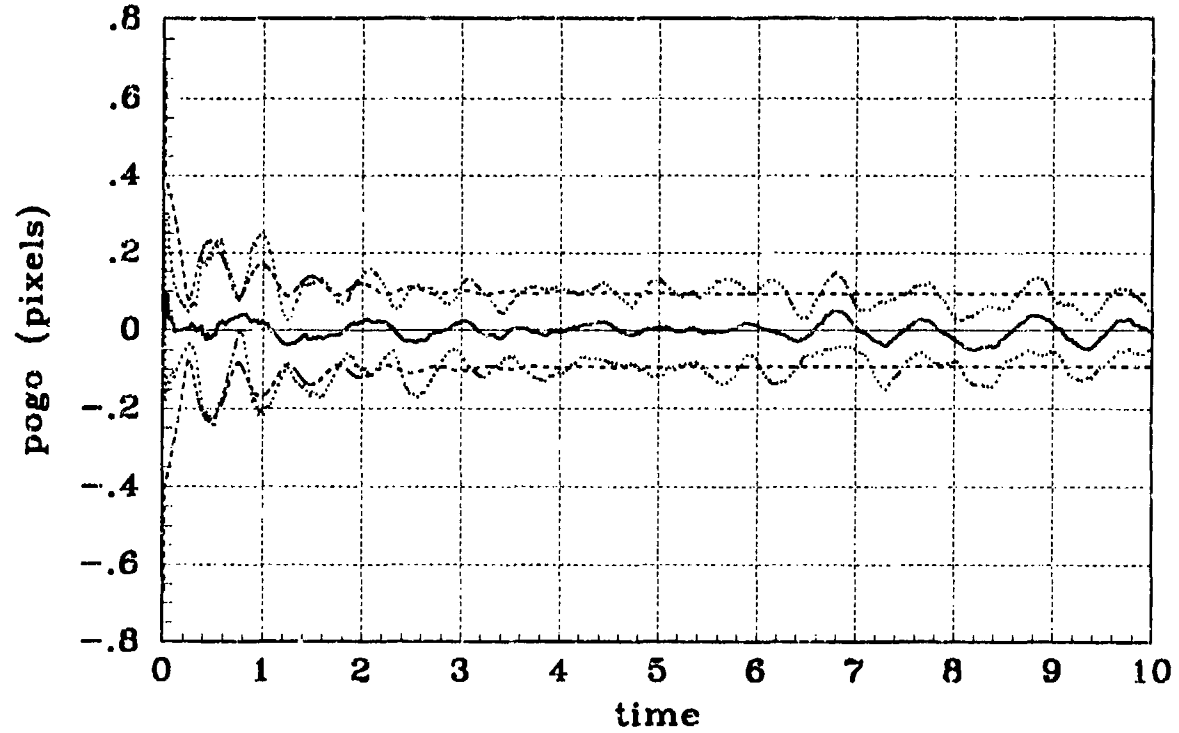


(b) Y JITTER STATISTICS - Filter #10, Pogo Cond 4

Figure C.36 Simulation #10, θ = 60°, Y-Atmospheric Jitter



(a) POGO OFFSET – Filter #10, Pogo Cond 4



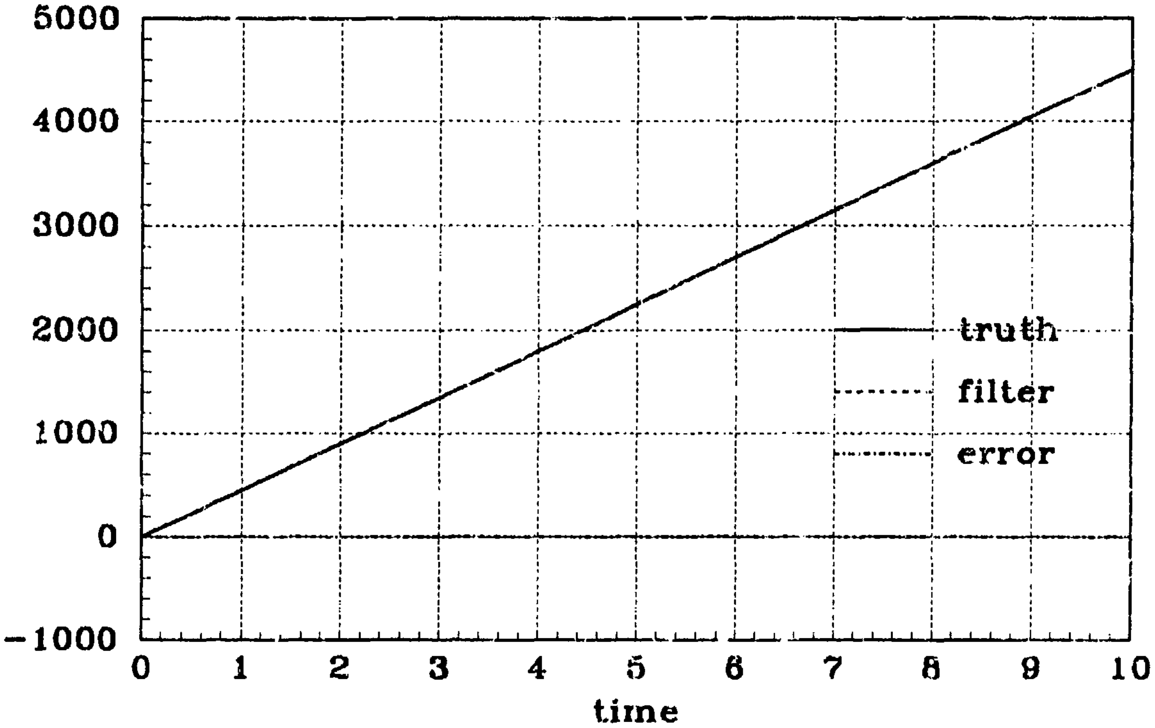
(b) POGO OFFSET STATISTICS – Filter #10, Pogo Cond 4

Figure C.37 Simulation #10, θ = 60°, Pogo Offset

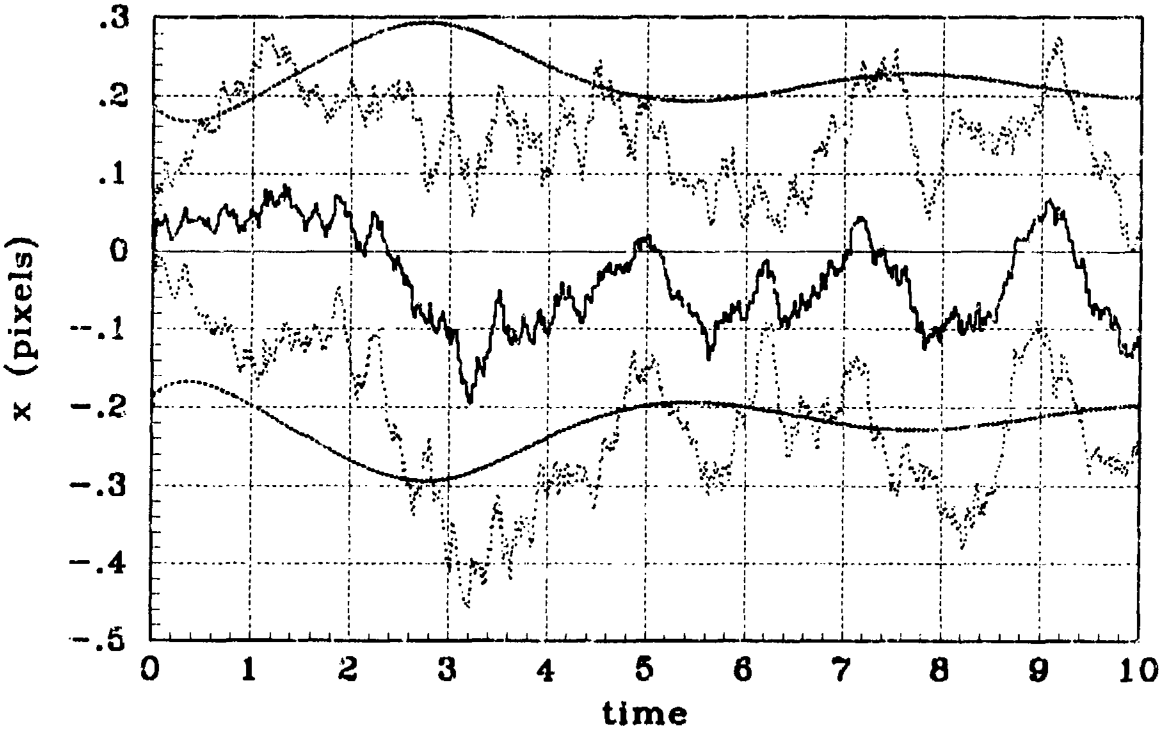
Appendix D

MSOFE Analysis Using Varying Pogo Conditions

This appendix contains the average states and error statistic plots for the MSOFE simulation #10 using a general angle model (nonlinear measurement update) for the different plume pogo characteristics listed in Table 5.3. Data collection and symbology depicted in the two types of plots contained in this appendix are explained in Appendices A and B. The state comparison plots show the ensemble average truth state over the 10 Monte Carlo runs compared to the same statistic for the filter estimate. The error statistics plots represent the error mean and mean ±1 standard deviation values in pixels of the errors between the filter-estimated and true state.

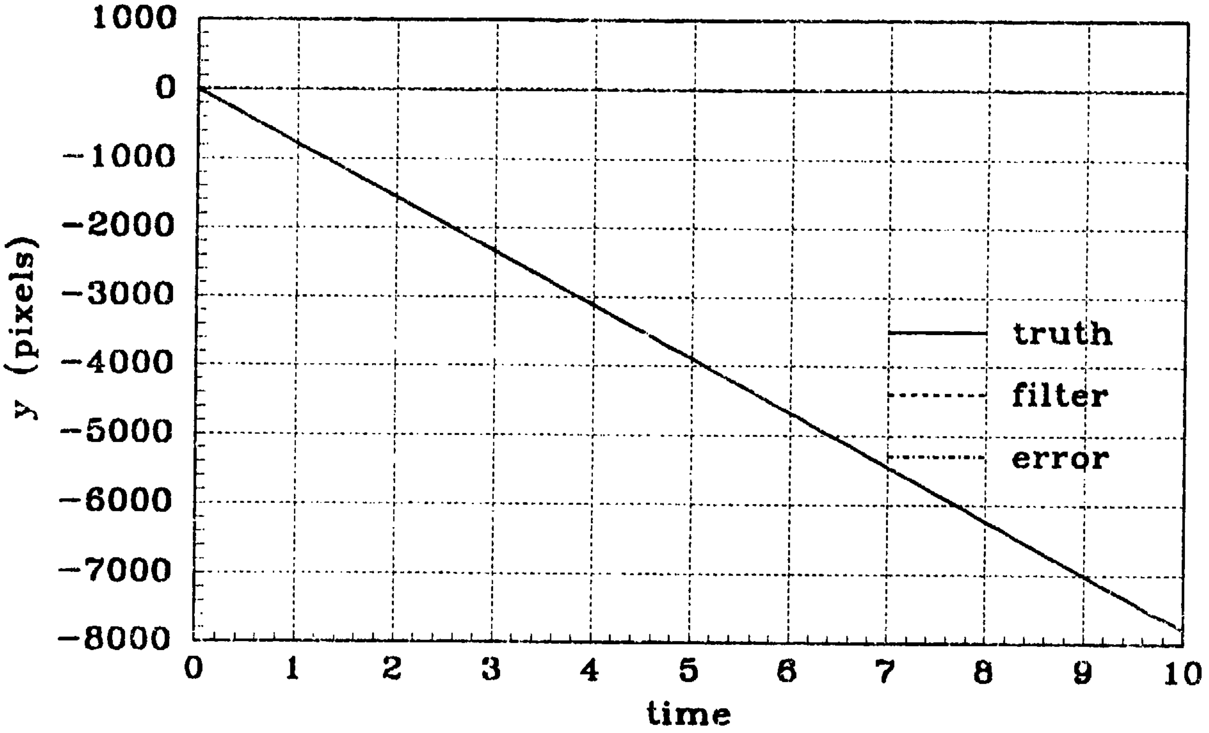


(a) X POSITION – Filter #10,Nonlinear, PC#1

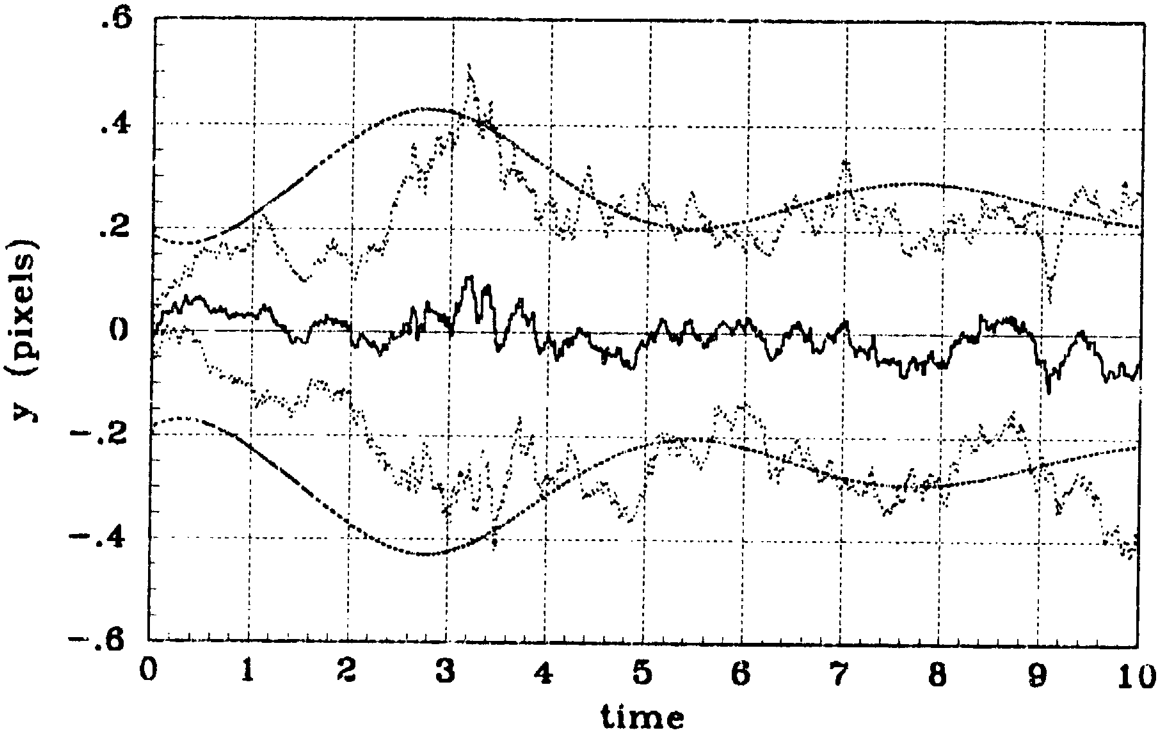


(b) X POSITION STATISTICS – Filter #10,Nonlinear, PC#1

Figure D.1 Pogo Condition #1, Simulation #10, General Angle, X-Position

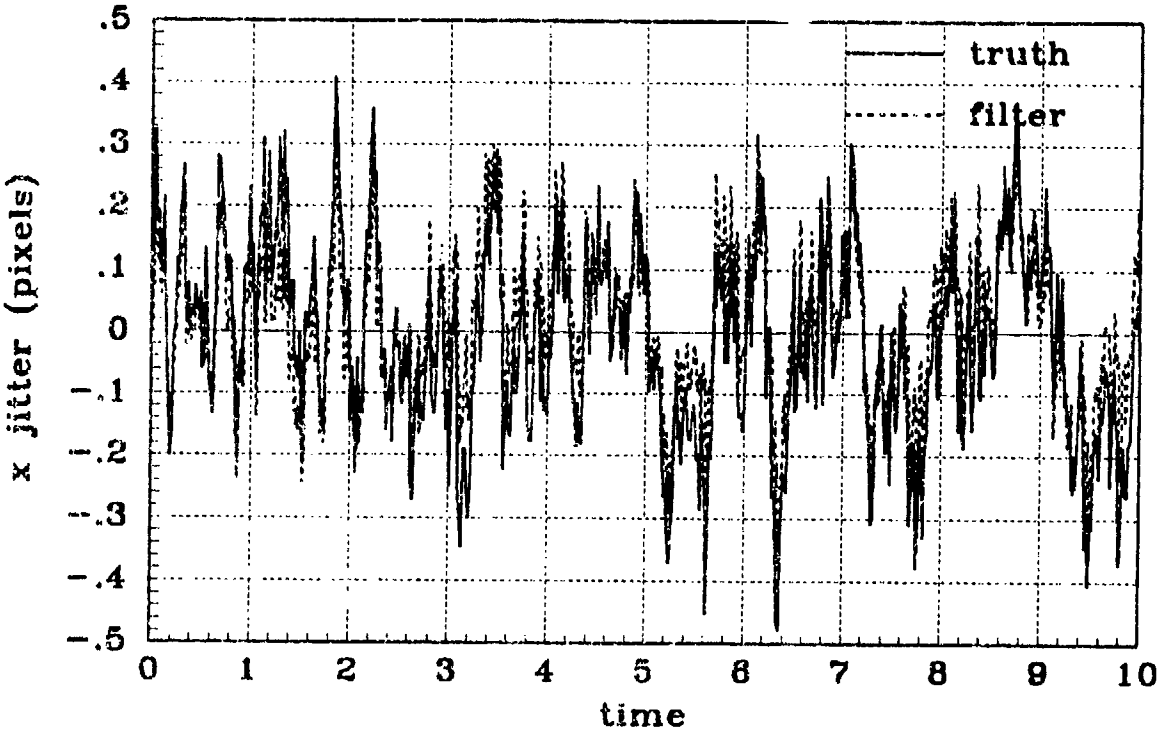


(a) Y POSITION – Filter #10,Nonlinear, PC#1

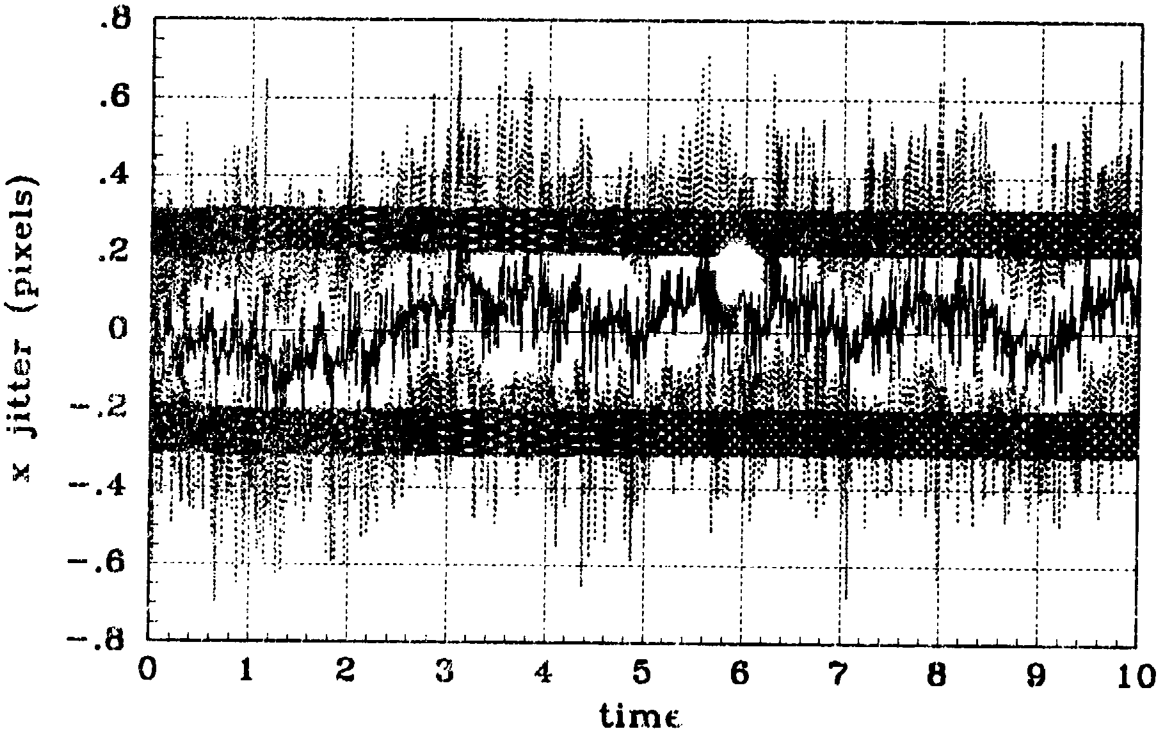


(b) Y POSITION STATISTICS – Filter #10,Nonlinear, PC#1

Figure D.2 Pogo Condition #1, Simulation #10, General Angle, Y-Position

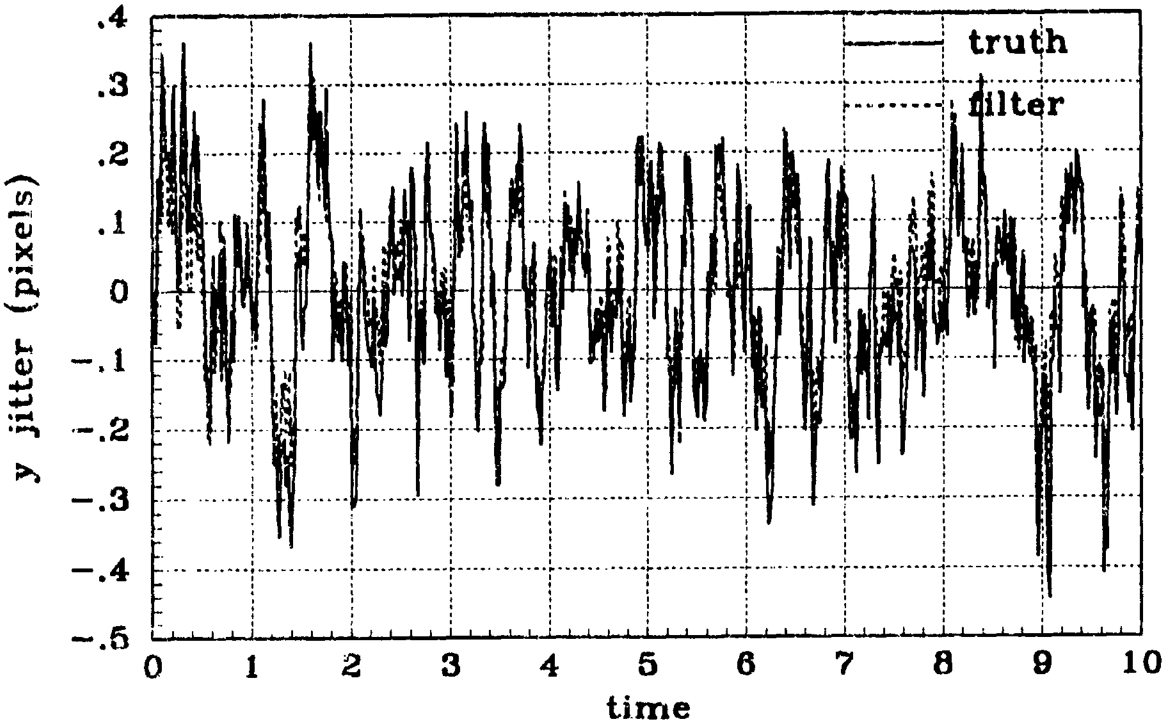


(a) X JITTER - Filter #10,Nonlinear, PC#1

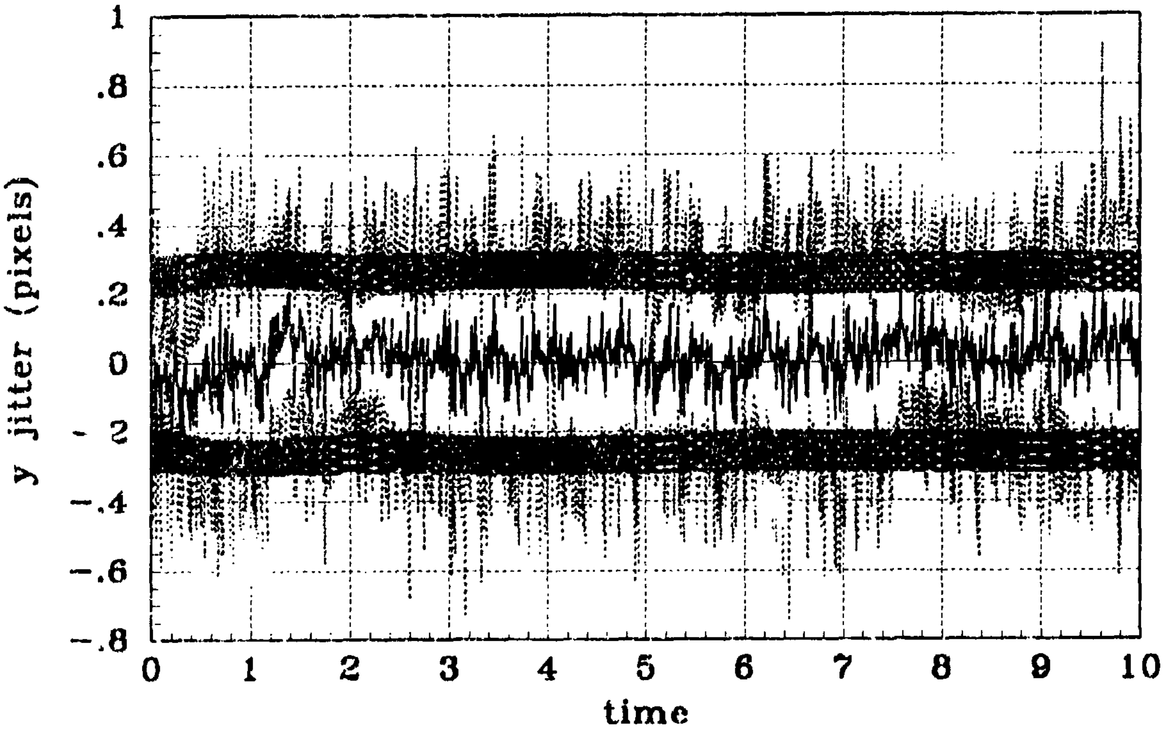


(b) X JITTER STATISTICS - Filter #10,Nonlinear, PC#1

Figure D.3 Pogo Condition #1, Simulation #10, General Angle, X-Atmospheric Jitter

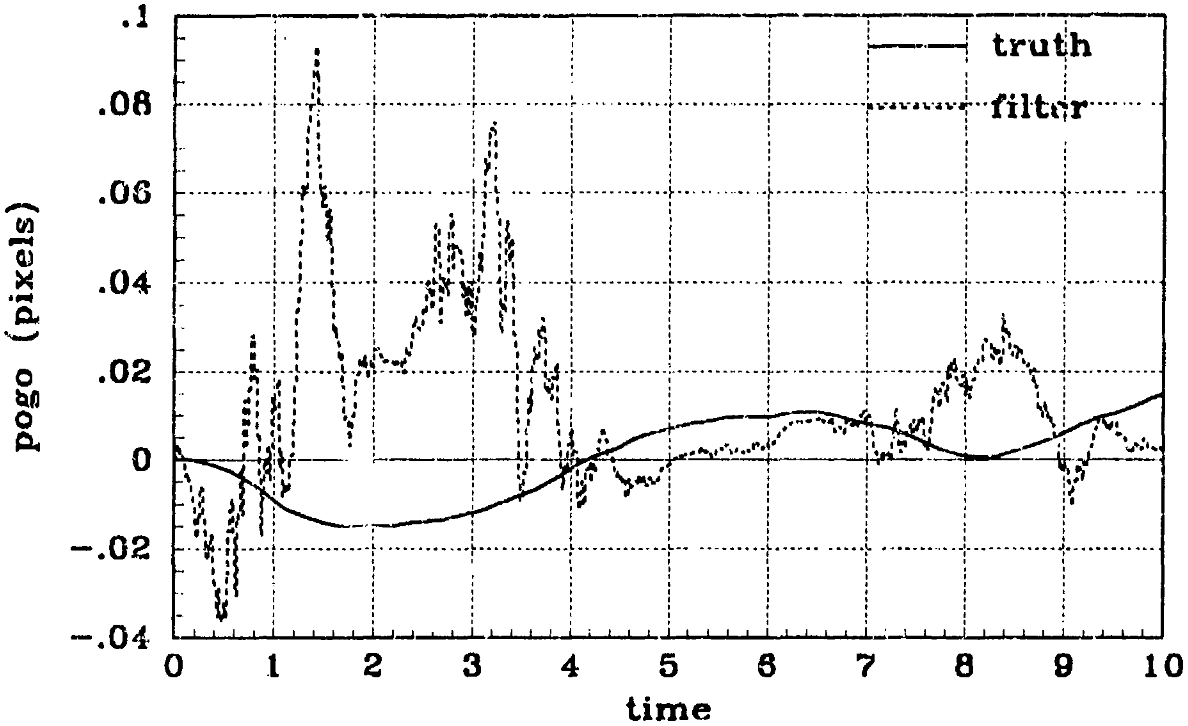


(a) Y JITTER – Filter #10,Nonlinear, PC#1

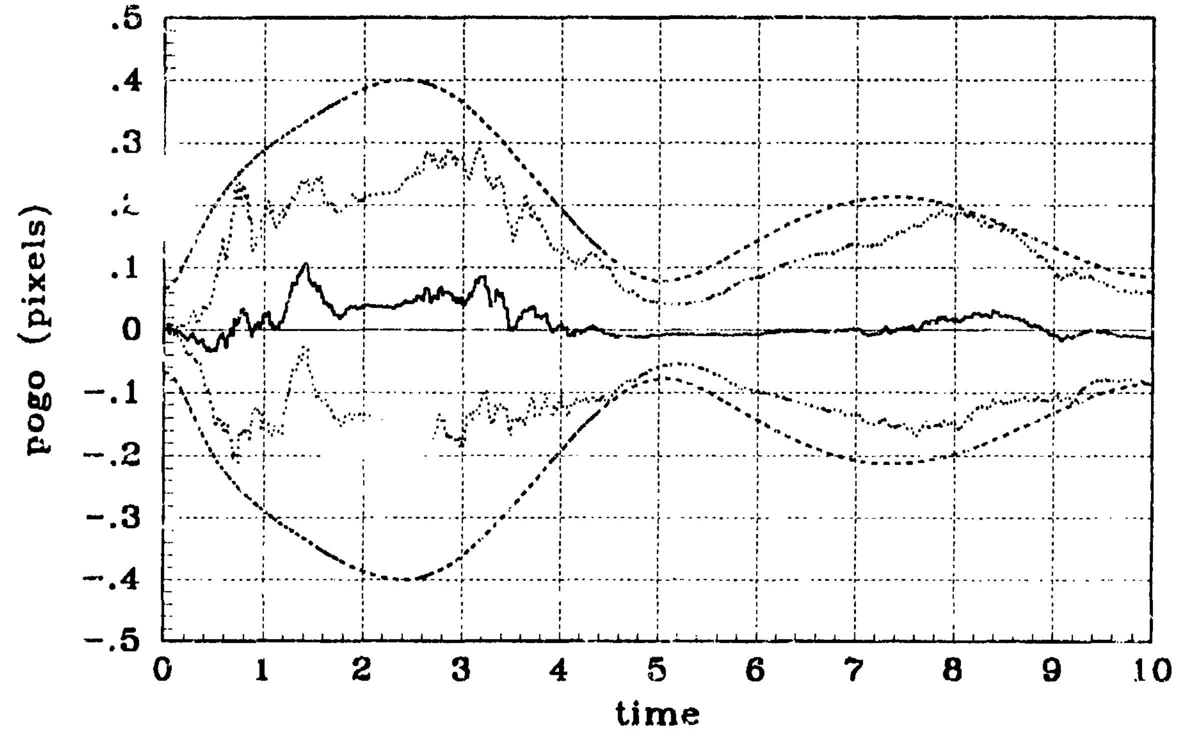


(b) Y JITTER STATISTICS – Filter #10,Nonlinear, PC#1

Figure D.4 Pogo Condition #1, Simulation #10, General Angle, Y-Atmospheric Jitter

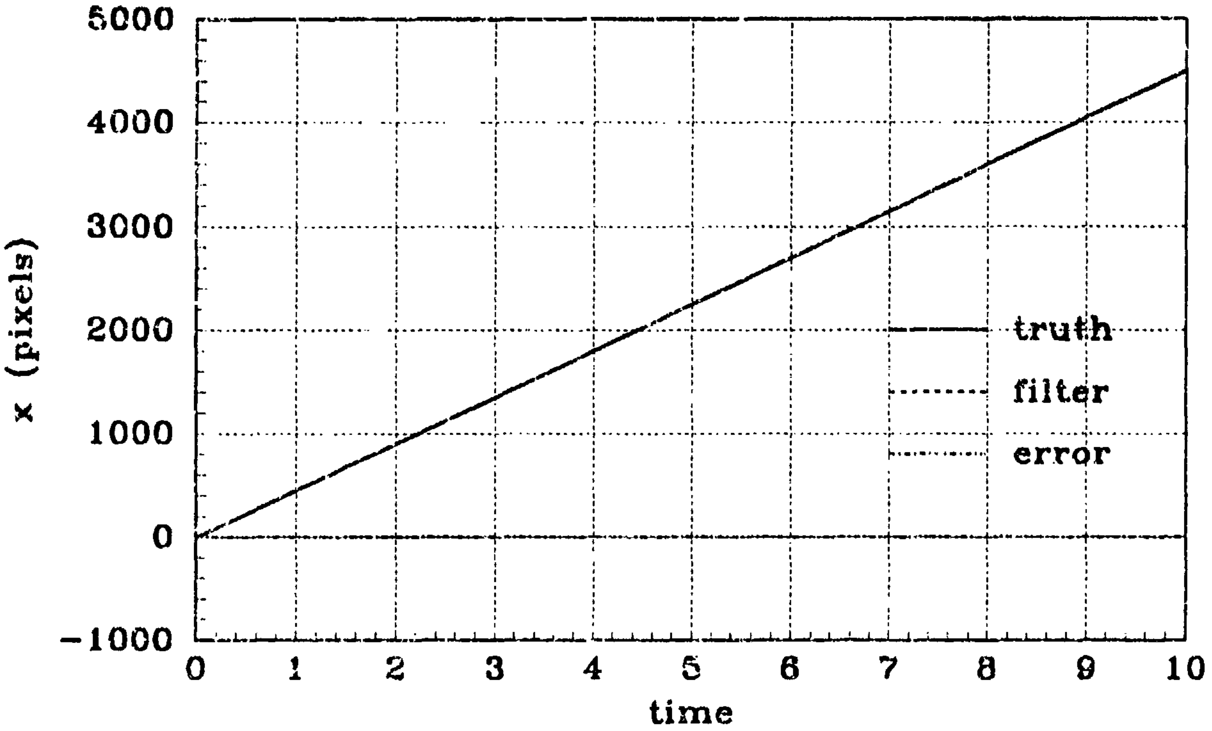


(a) POGO OFFSET - Filter #10,Nonlinear, PC#1

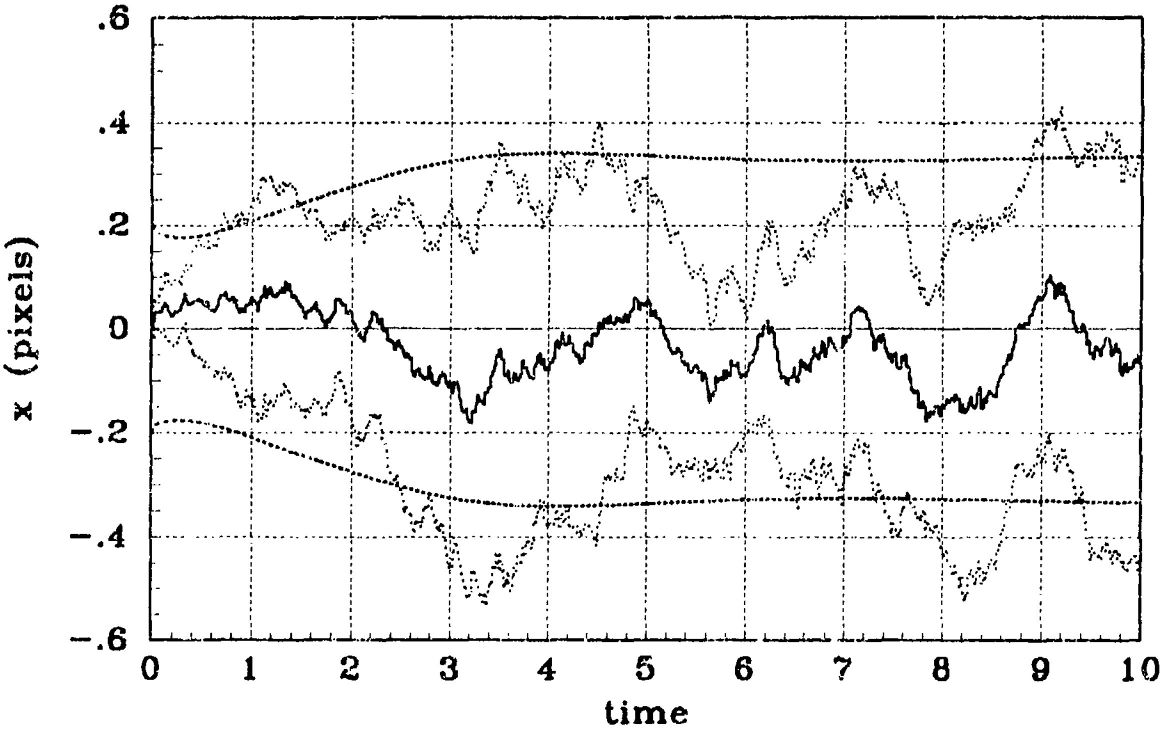


(b) POGO OFFSET STATISTICS - Filter #10,Nonlinear, PC#1

Figure D.5 Pogo Condition #1, Simulation #10, General Angle, Pogo Offset

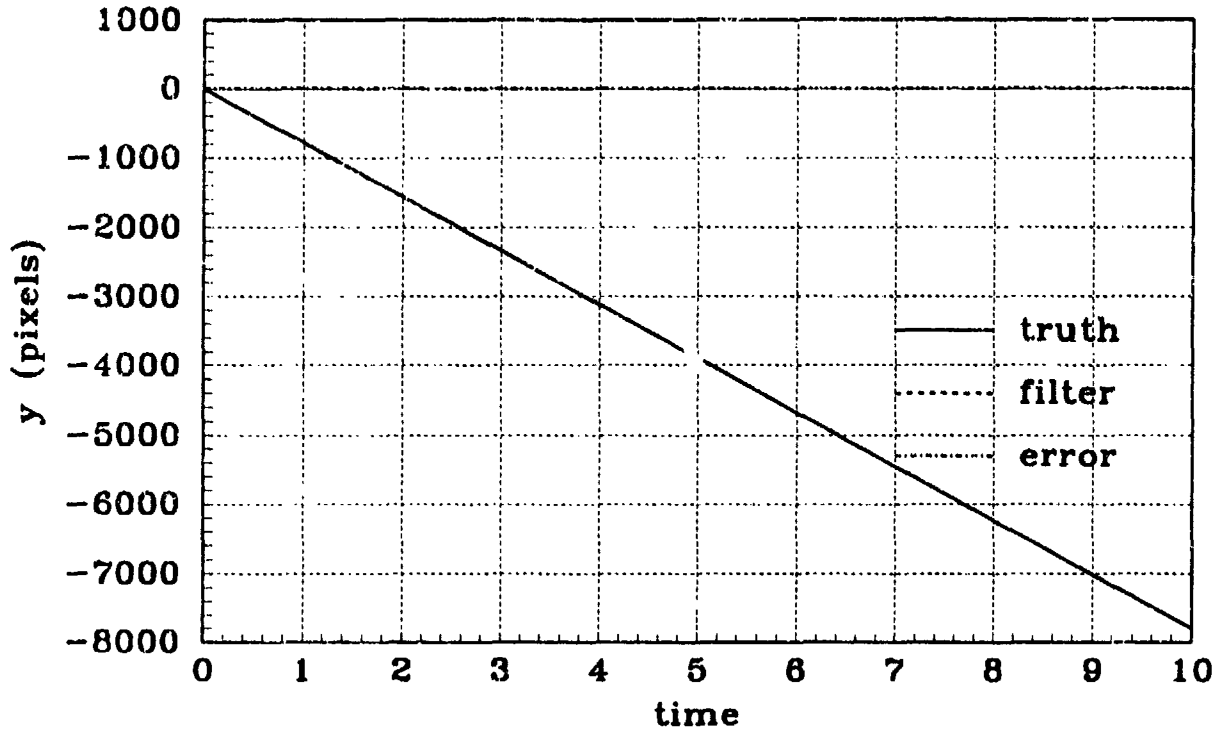


(a) X POSITION – Filter #10,Nonlinear, PC#2

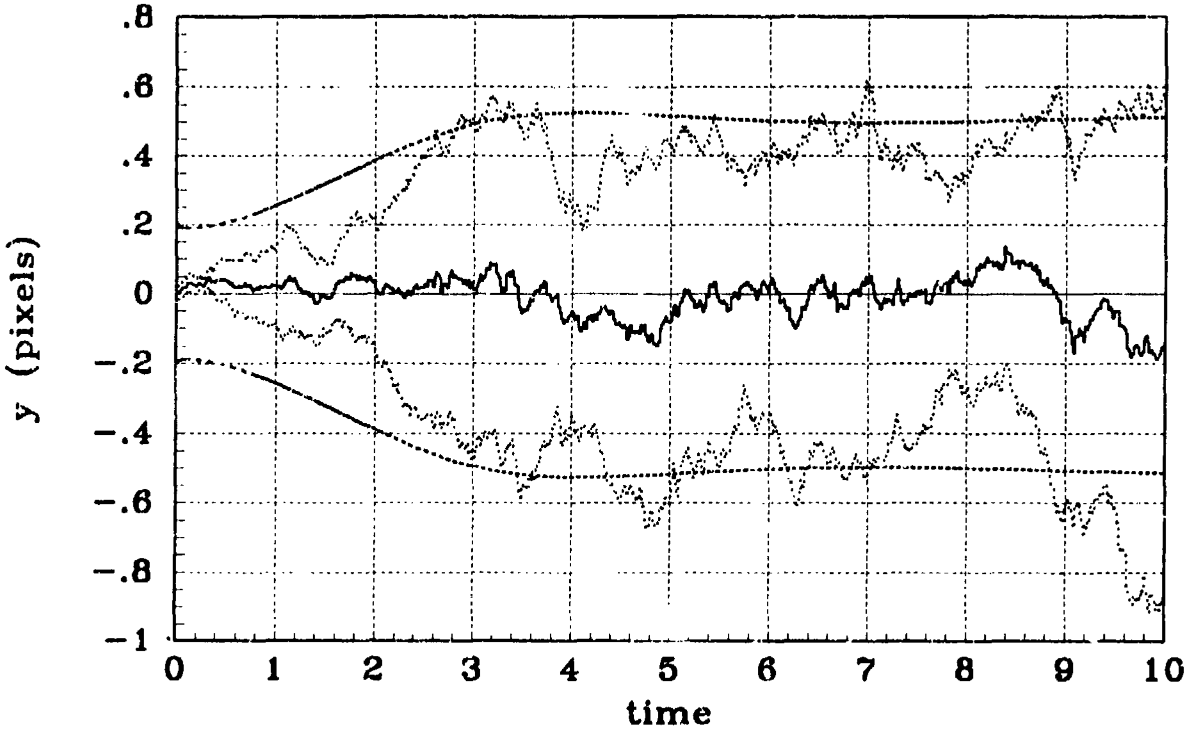


(b) X POSITION STATISTICS – Filter #10,Nonlinear, PC#2

Figure D 6 Pogo Condition #2, Simulation #10, General Angle, X-Position

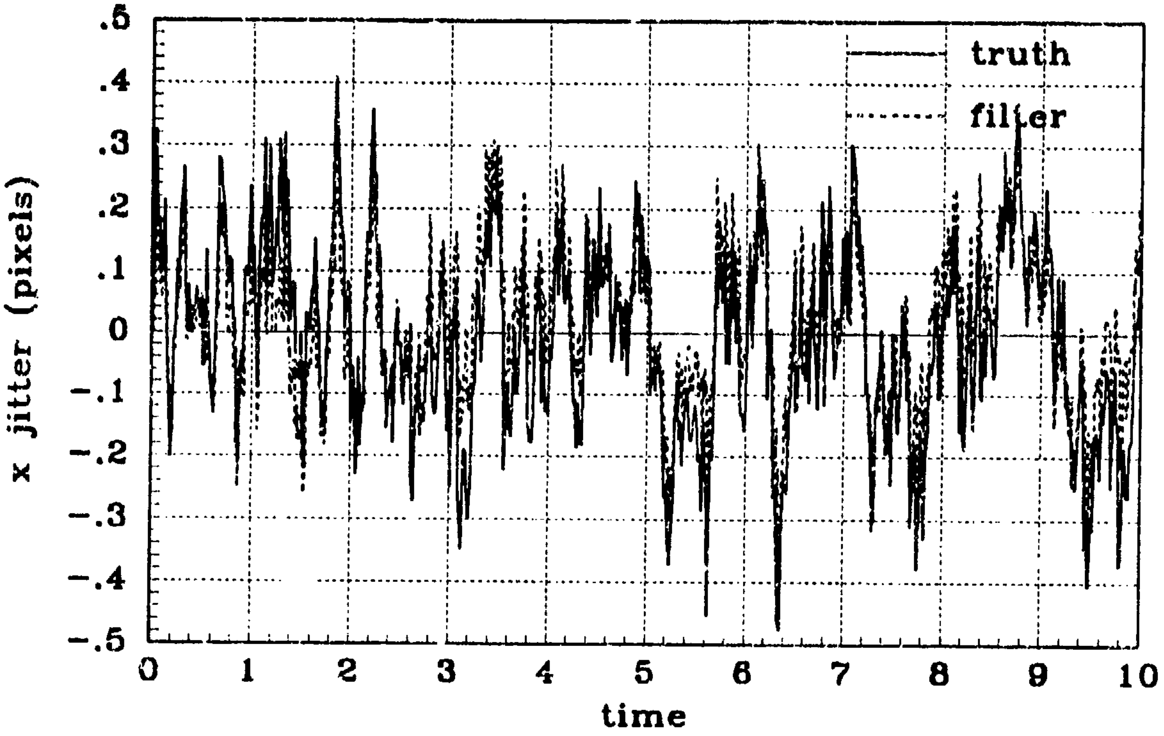


(a) Y POSITION - Filter #10,Nonlinear, PC#2

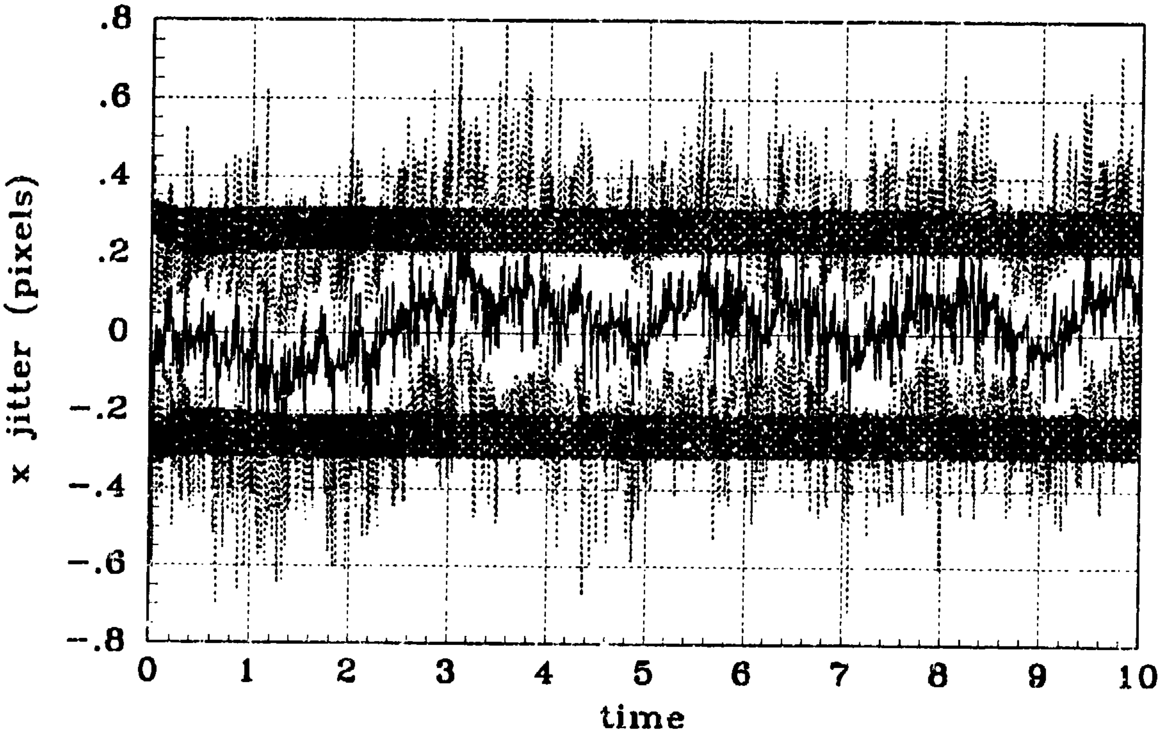


(b) Y POSITION STATISTICS - Filter #10,Nonlinear, PC#2

Figure D.7 Pogo Condition #2, Simulation #10, General Angle, Y-Position

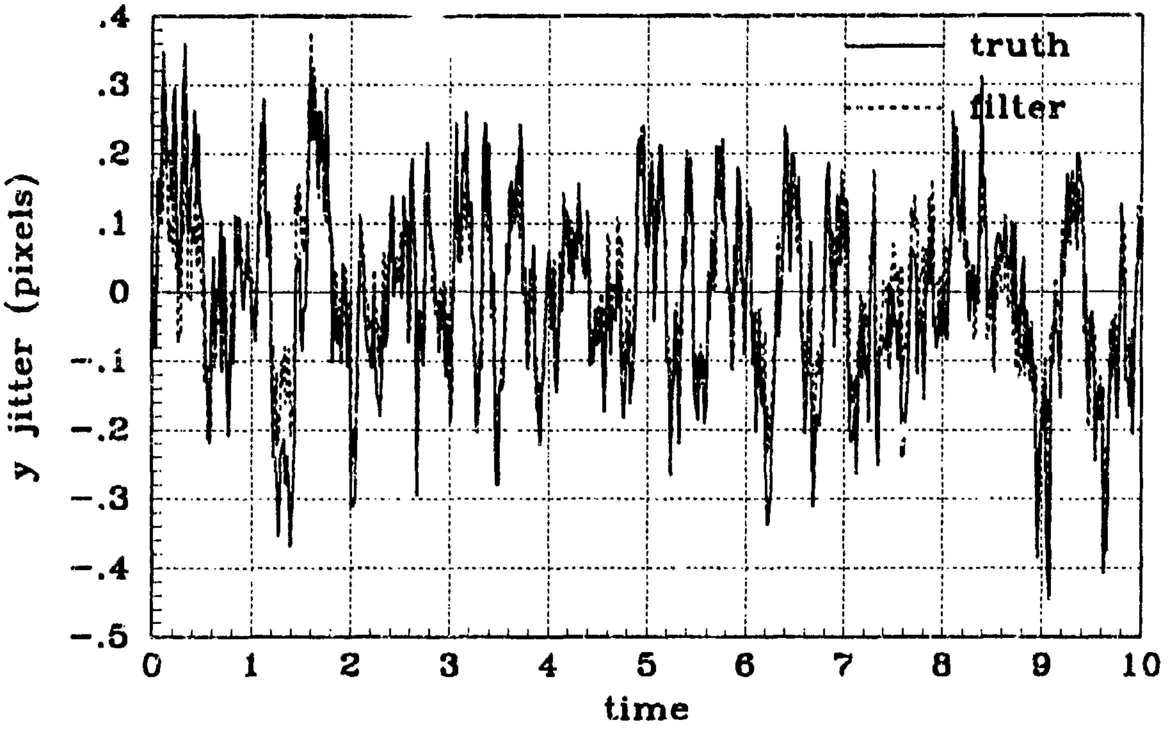


(a) X JITTER − Filter #10,Nonlinear, PC#2

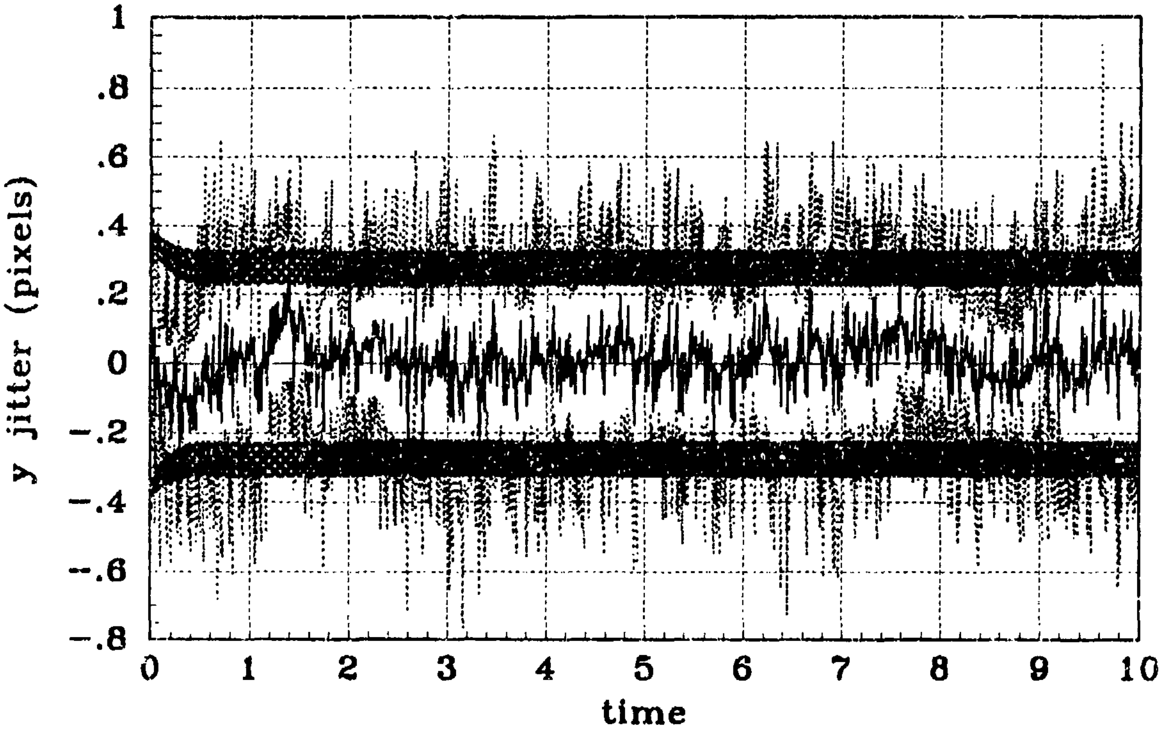


(b) X JITTER STATISTICS − Filter #10,Nonlinear, PC#2

Figure D.8 Pogo Condition #2, Simulation #10, General Angle, X-Atmospheric Jitter

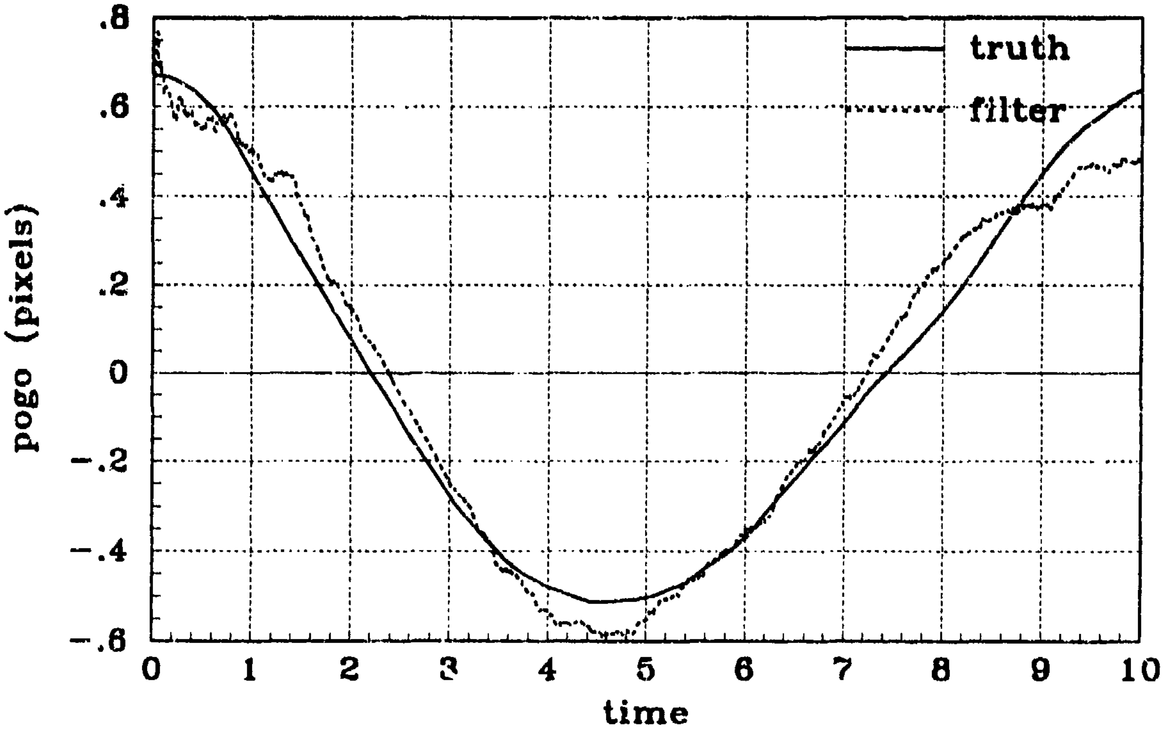


(a) Y JITTER - Filter #10,Nonlinear, PC#2

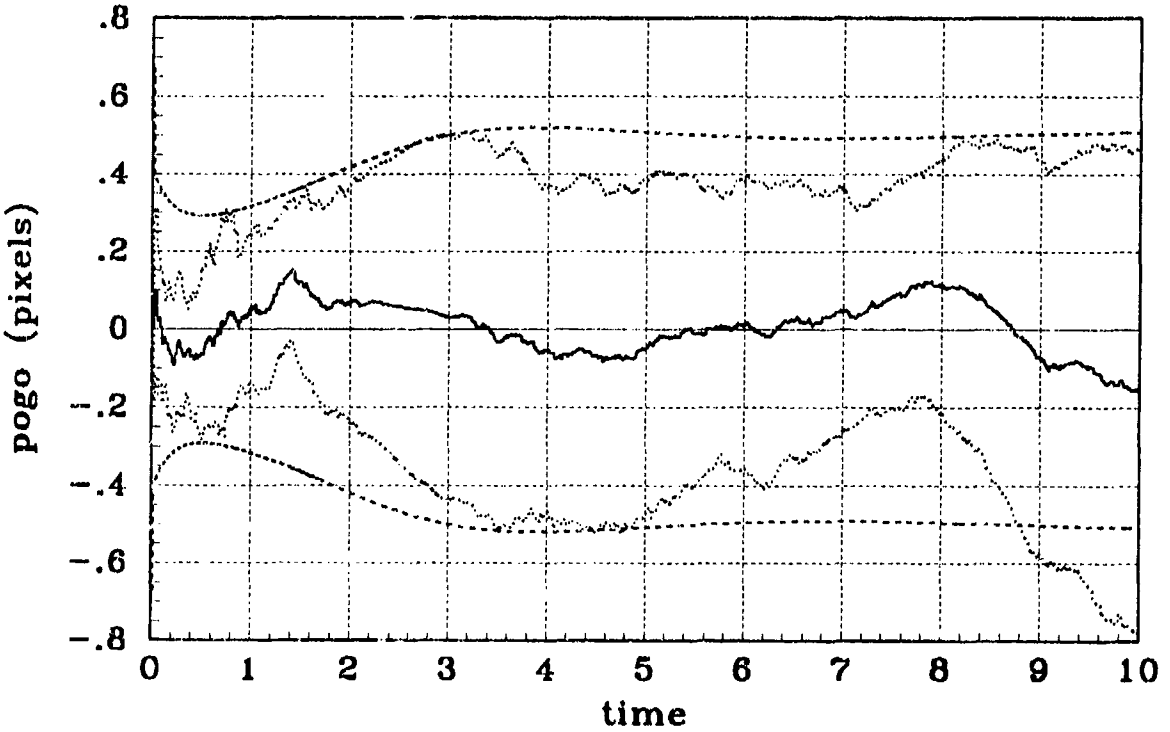


(b) Y JITTER STATISTICS - Filter #10,Nonlinear, PC#2

Figure D.9 Pogo Condition #2, Simulation #10, General Angle, Y-Atmospheric Jitter

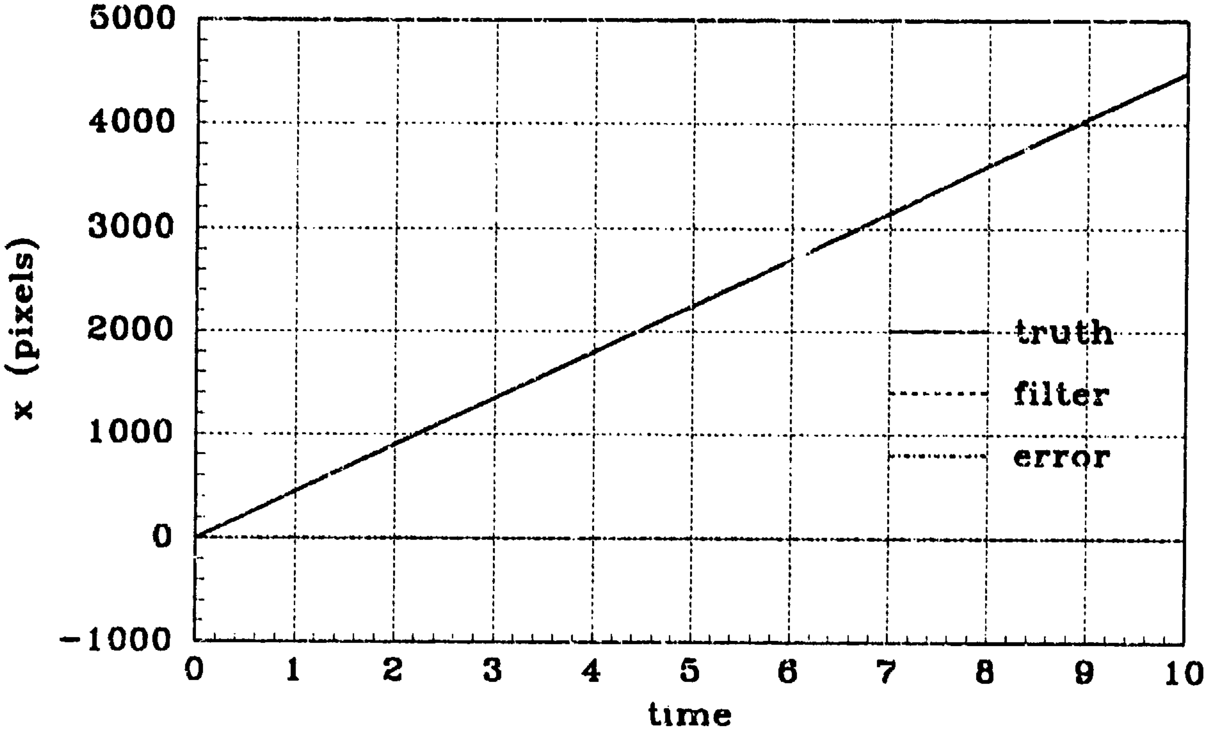


(a) POGO OFFSET - Filter #10,Nonlinear, PC#2

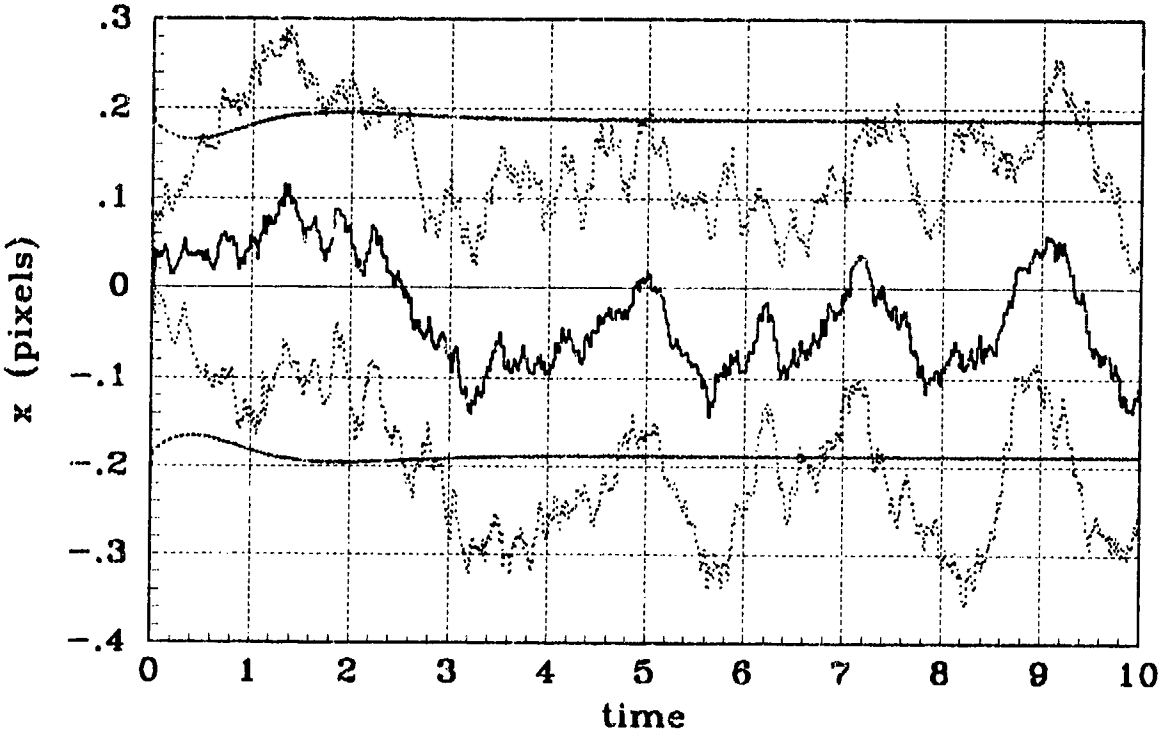


(b) POGO OFFSET STATISTICS - Filter #10,Nonlinear, PC#2

Figure D.10 Pogo Condition #2, Simulation #10, General Angle, Pogo Offset

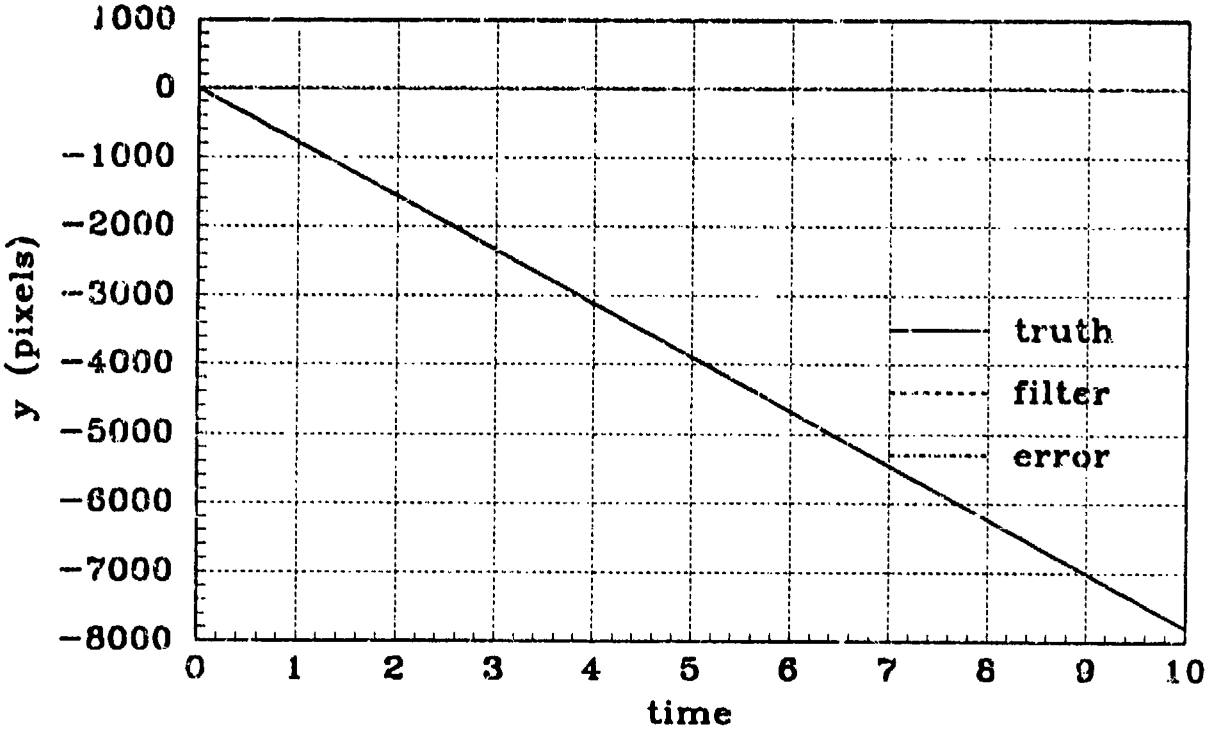


(a) X POSITION − Filter #10,Nonlinear, PC#3

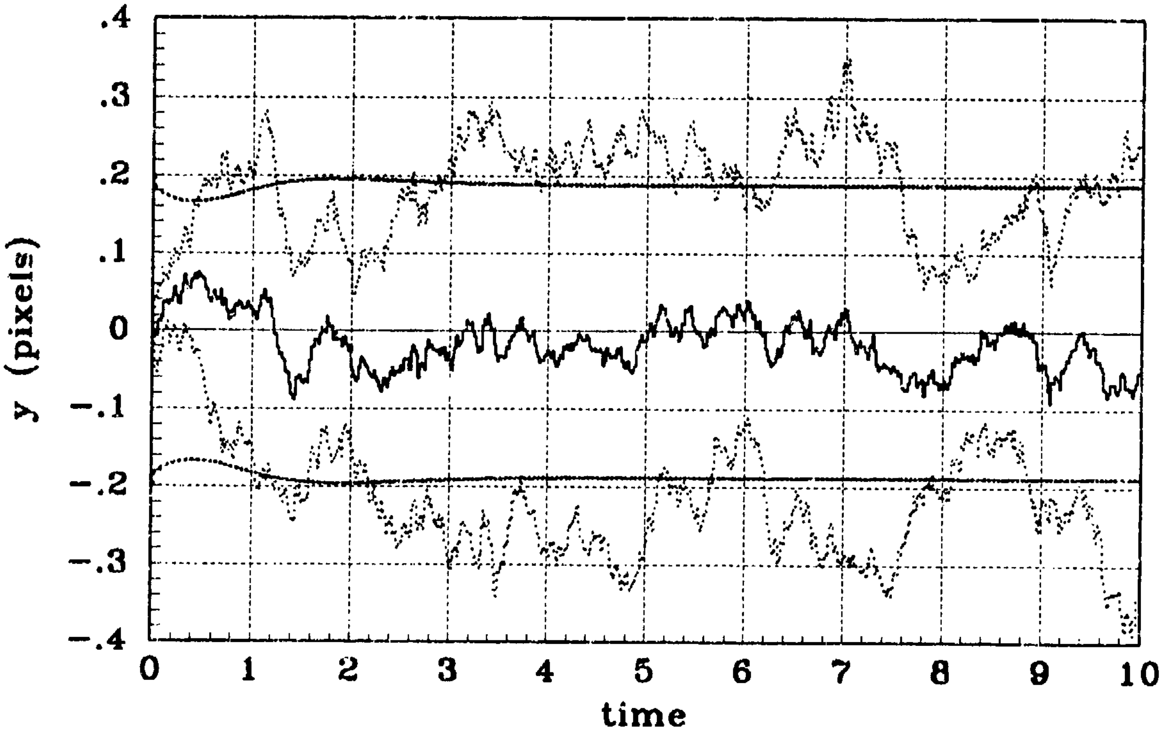


(b) X POSITION STATISTICS − Filter #10,Nonlinear, PC#3

Figure D.11 Pogo Condition #3, Simulation #10, General Angle, X-Position

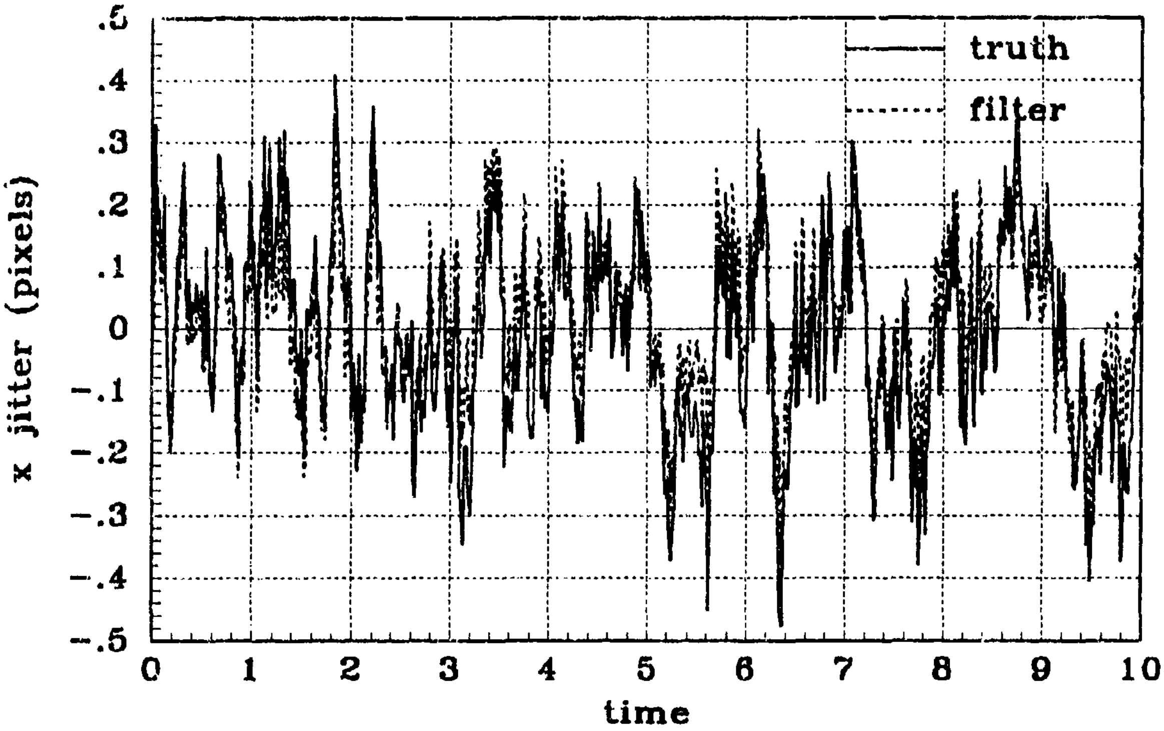


(a) Y POSITION – Filter #10,Nonlinear, PC#3

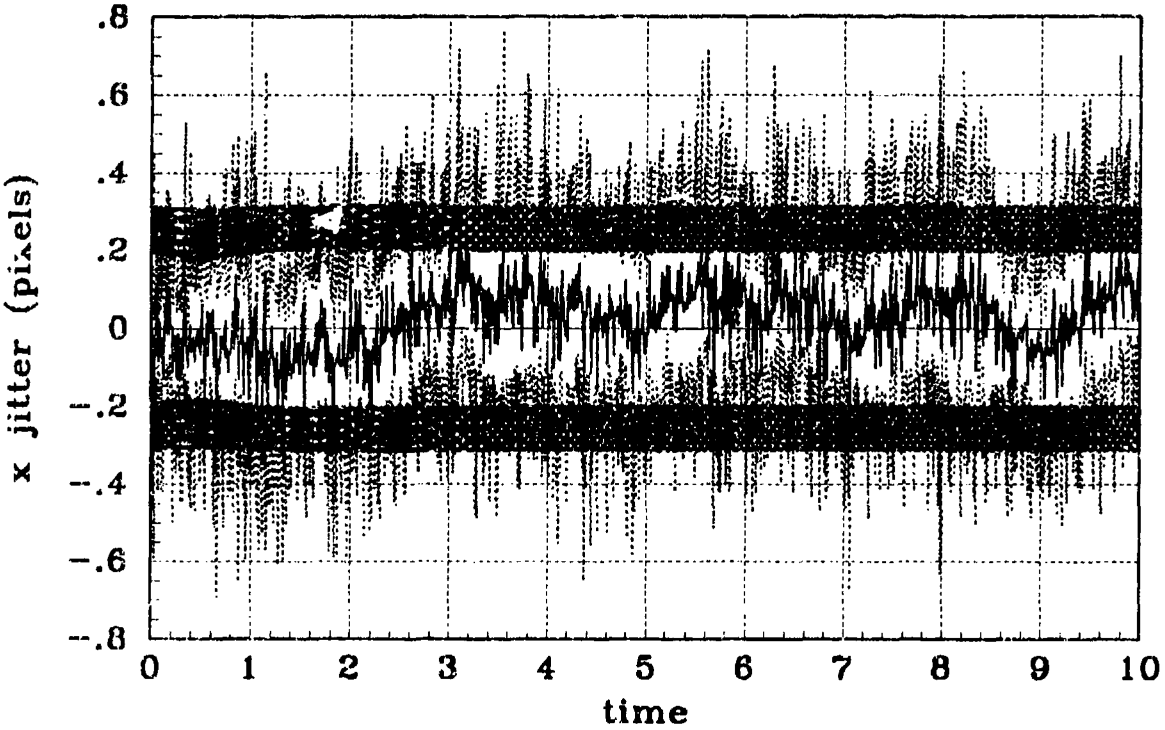


(b) Y POSITION STATISTICS – Filter #10,Nonlinear, PC#3

Figure D.12 Pogo Condition #3, Simulation #10, General Angle, Y-Position

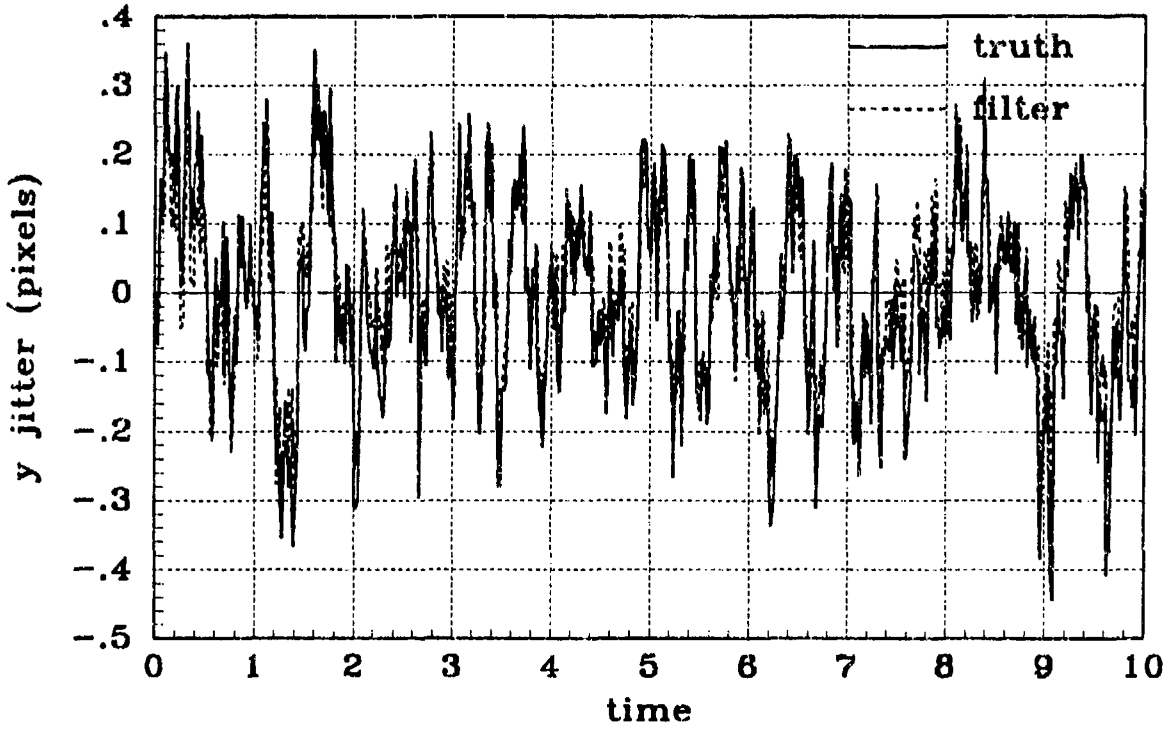


(a) X JITTER – Filter #10,Nonlinear, PC#3

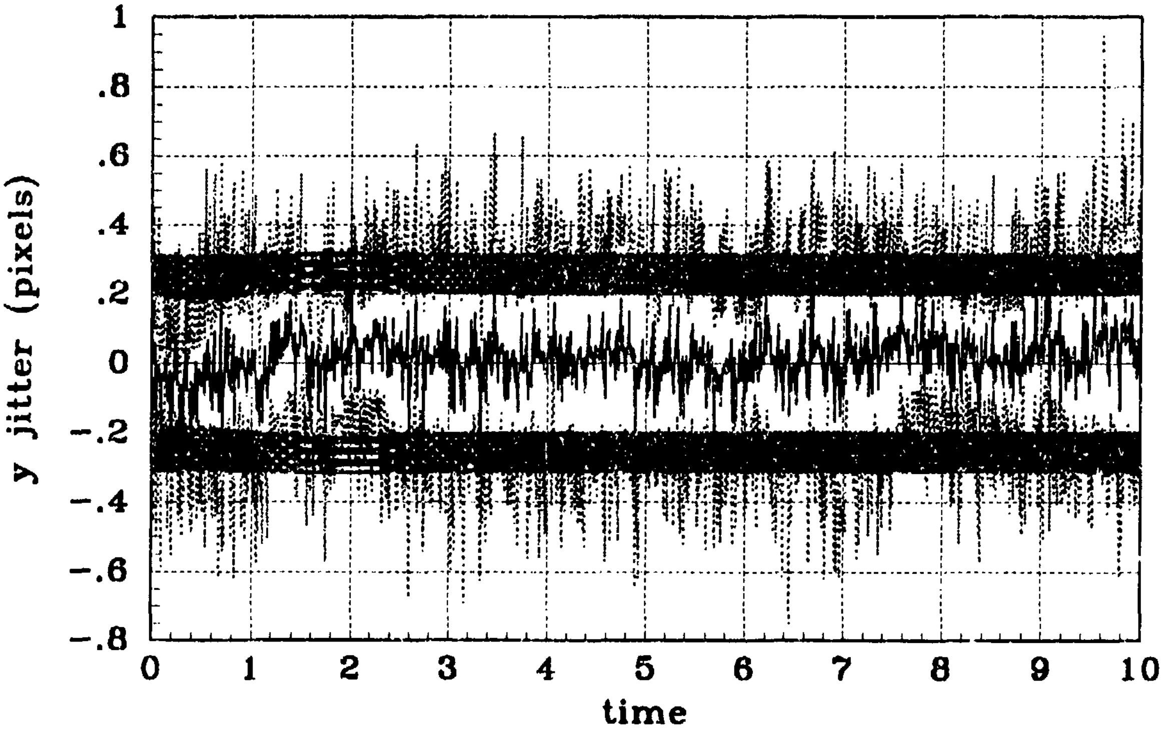


(b) X JITTER STATISTICS – Filter #10,Nonlinear, PC#3

Figure D.13 Pogo Condition #3, Simulation #10, General Angle, X-Atmospheric Jitter

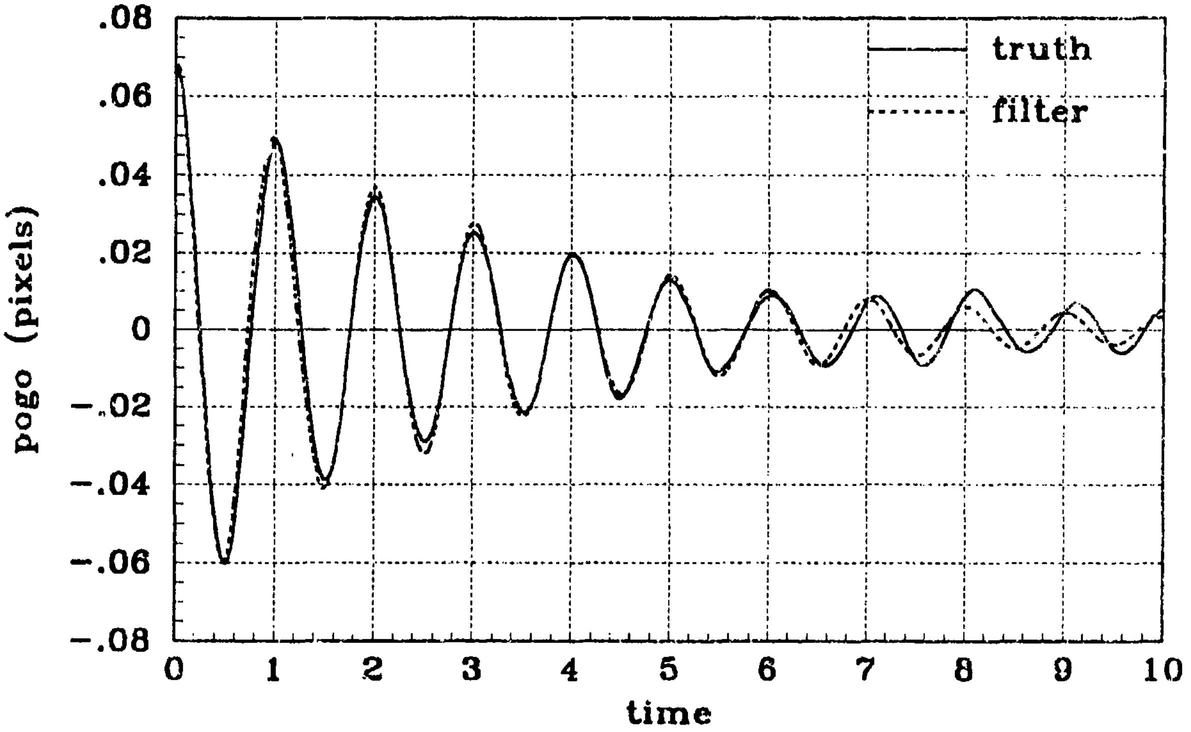


(a) Y JITTER − Filter #10,Nonlinear, PC#3

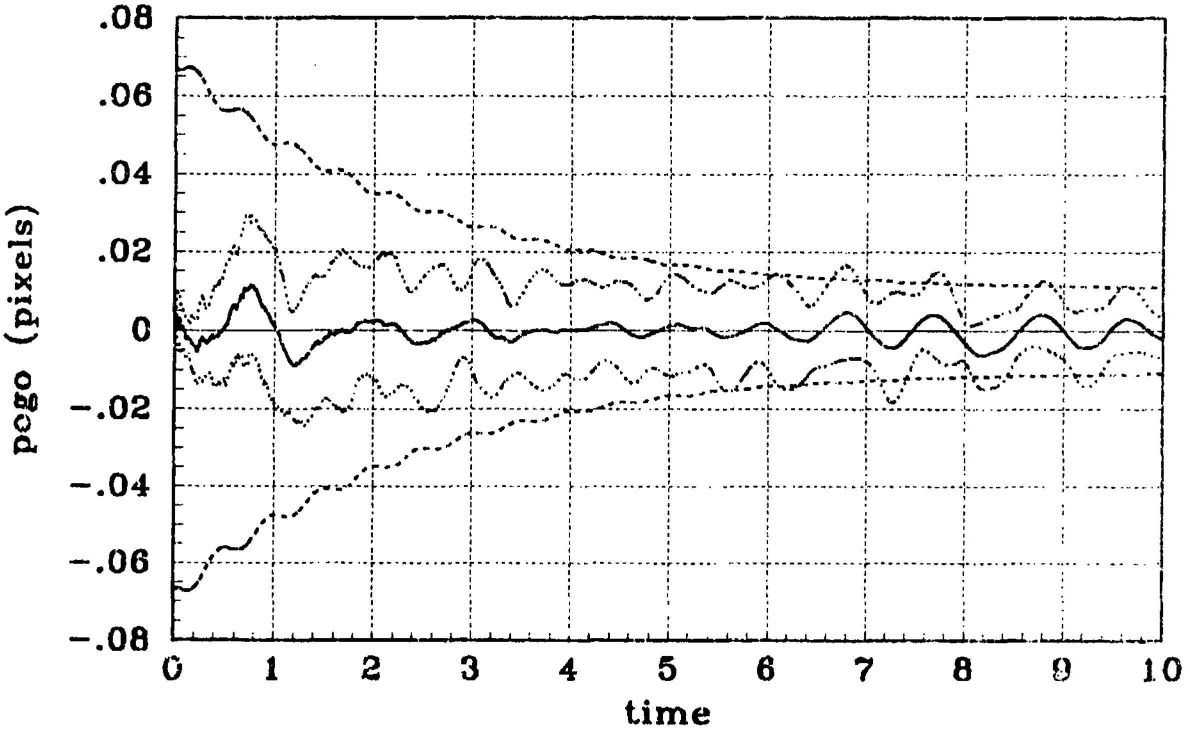


(b) Y JITTER STATISTICS − Filter #10,Nonlinear, PC#3

Figure D.14 Pogo Condition #3, Simulation #10, General Angle, Y-Atmospheric Jitter

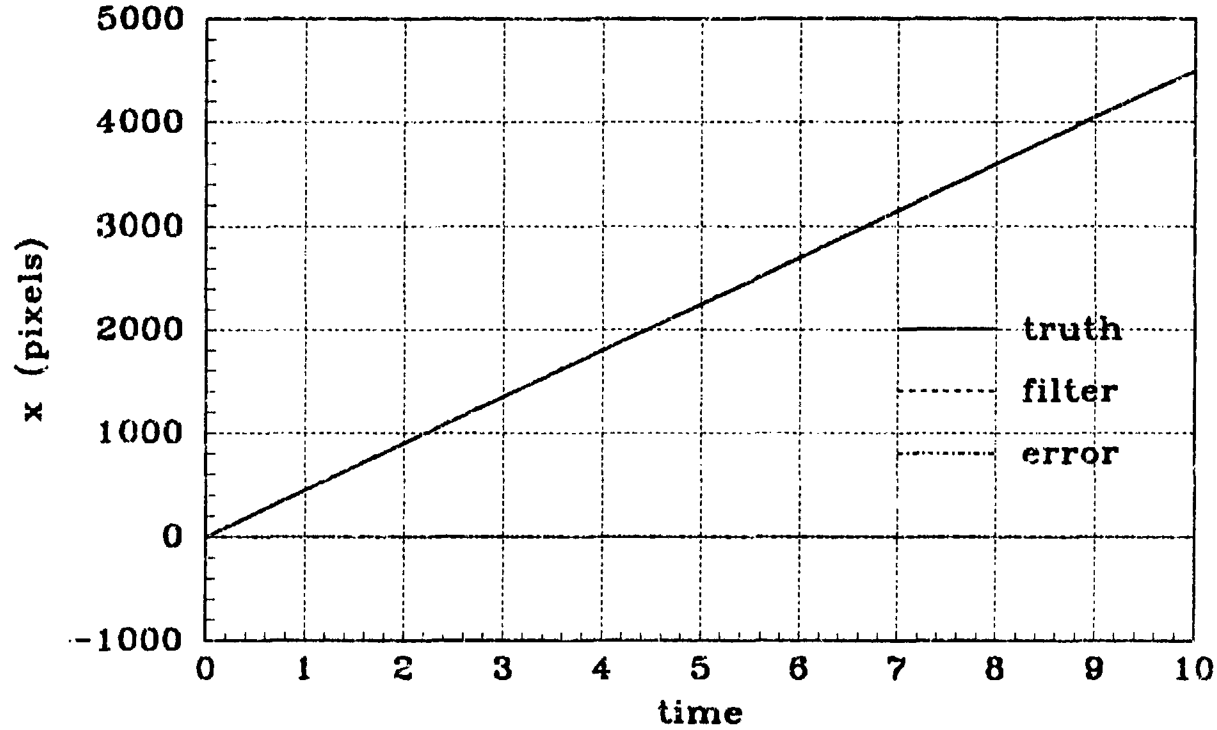


(a) POGO OFFSET – Filter #10,Nonlinear, PC#3

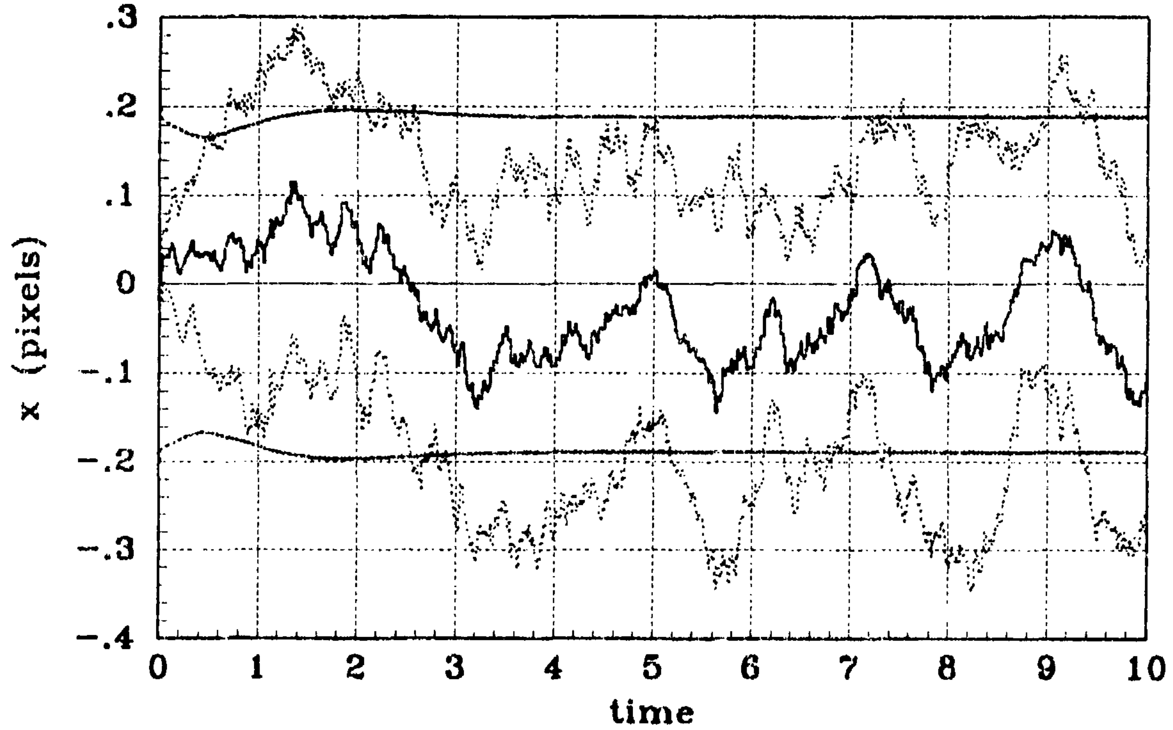


(b) POGO OFFSET STATISTICS – Filter #10,Nonlinear, PC#3

Figure D.15 Pogo Condition #3, Simulation #10, General Angle, Pogo Offset

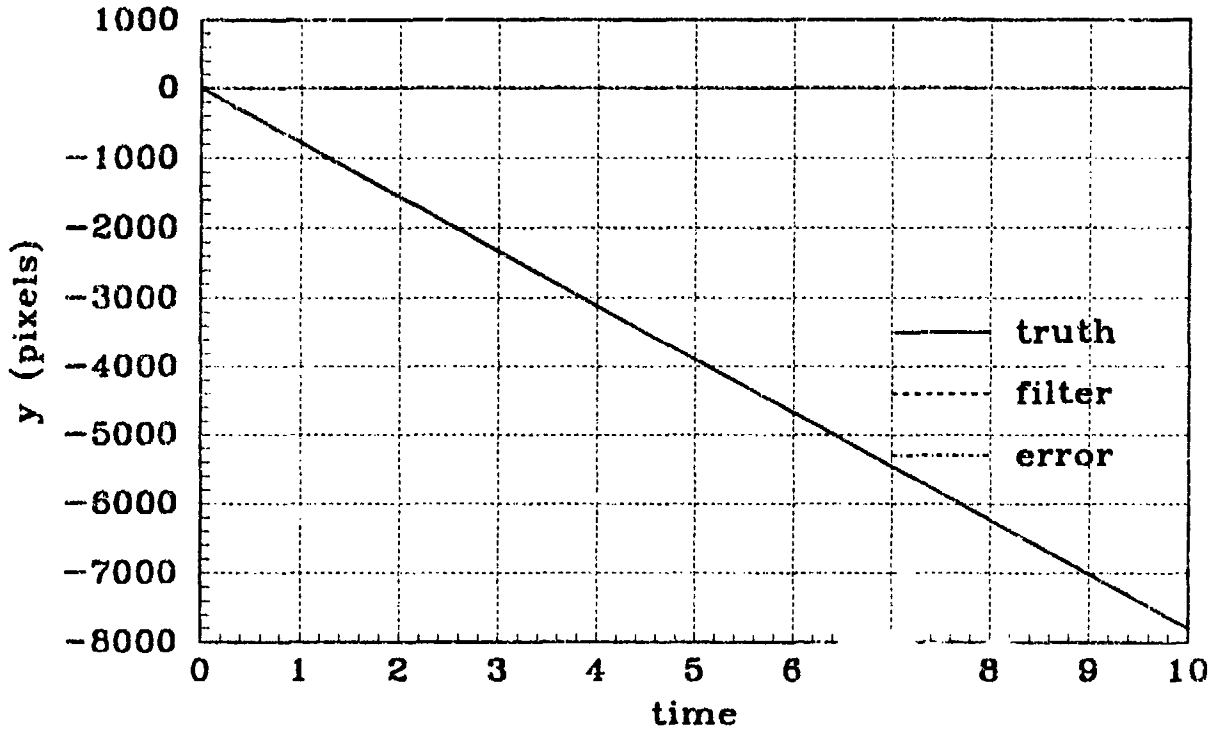


(a) X POSITION - Filter #10,Nonlinear, PC#4

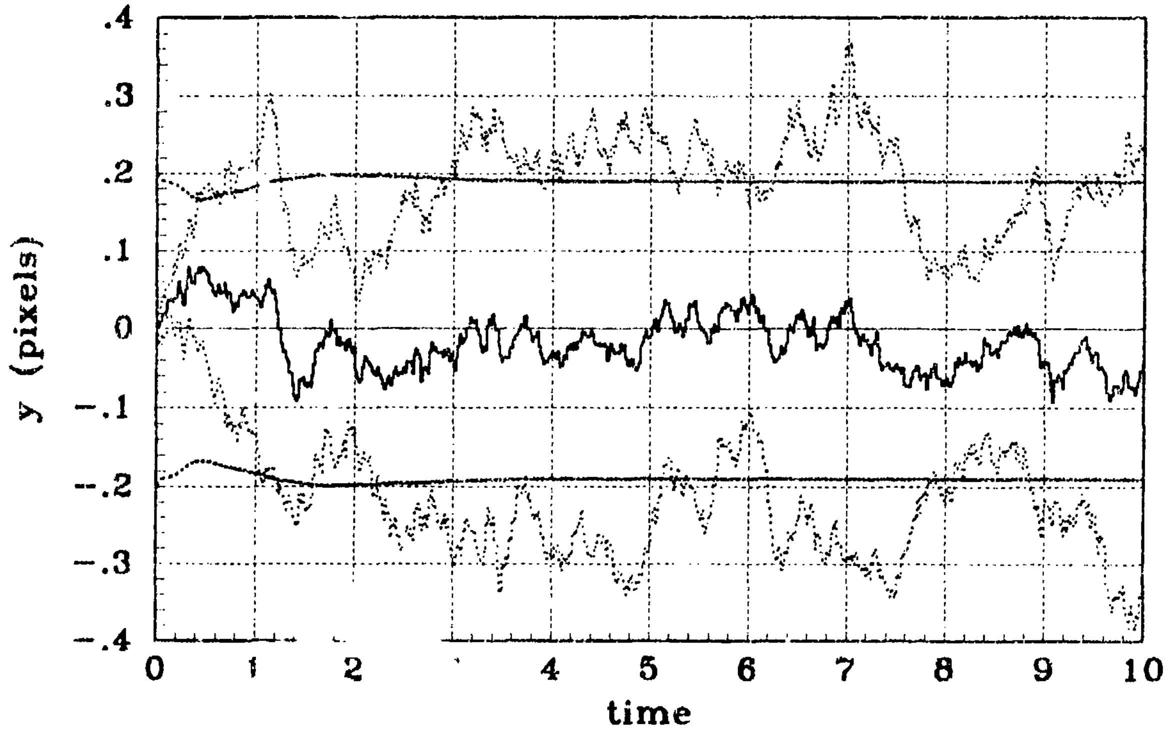


(b) X POSITION STATISTICS - Filter #10,Nonlinear, PC#4

Figure D.16 Pogo Condition #4, Simulation #10, General Angle, X-Position

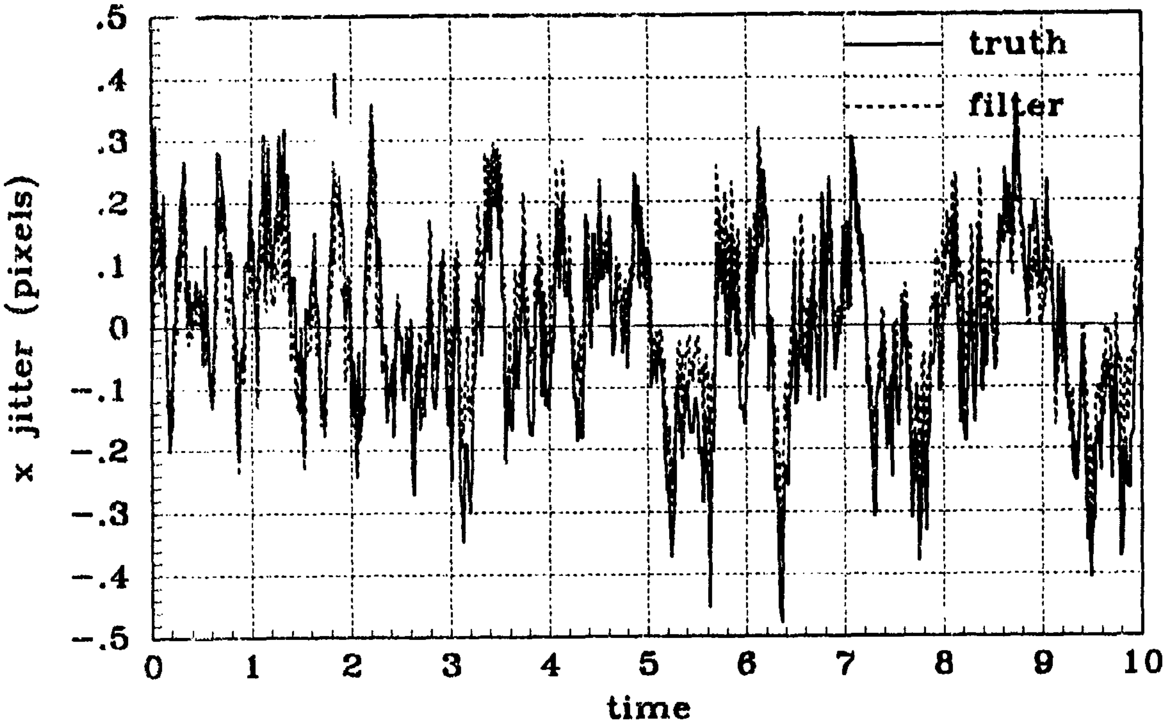


(a) Y POSITION – Filter #10,Nonlinear, PC#4

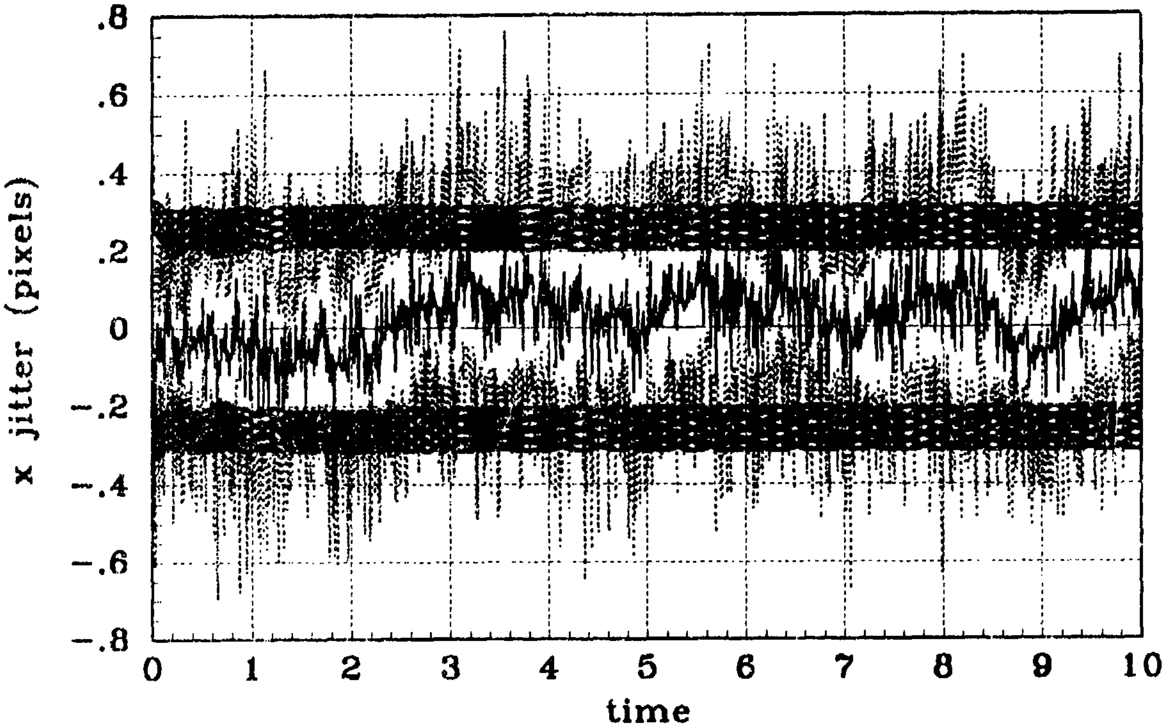


(b) Y POSITION STATISTICS – Filter #10,Nonlinear, PC#4

Figure D.17 Pogo Condition #4, Simulation #10, General Angle, Y-Position

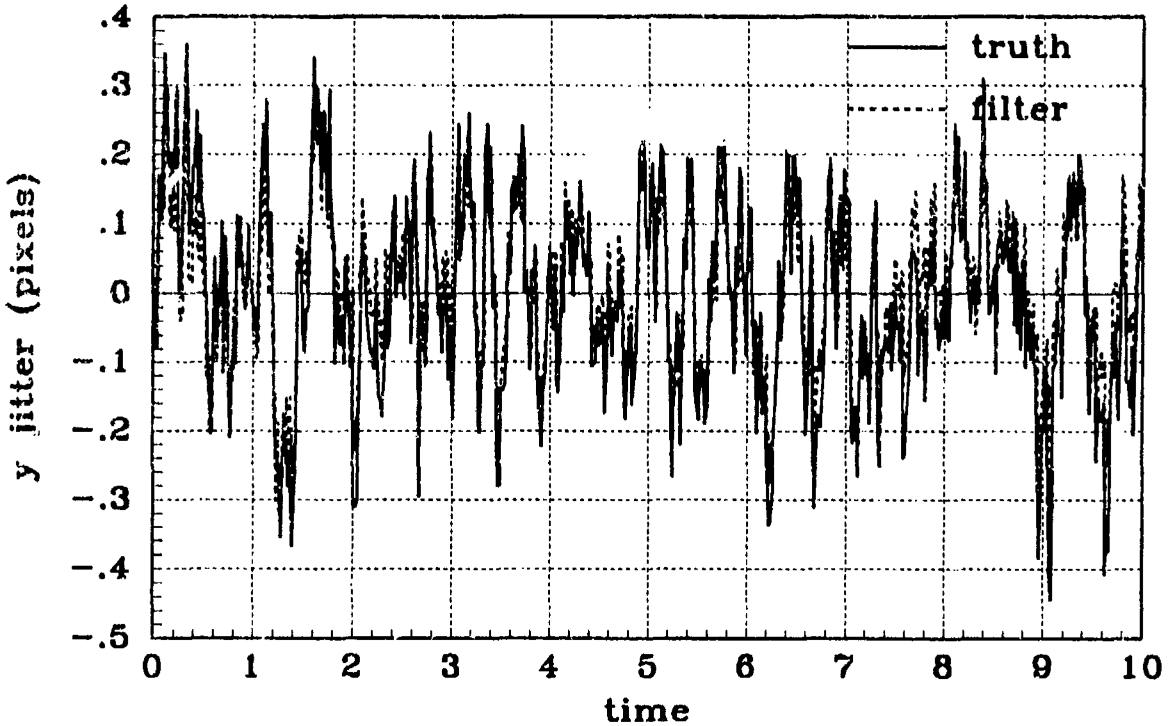


(a) X JITTER - Filter #10,Nonlinear, PC#4

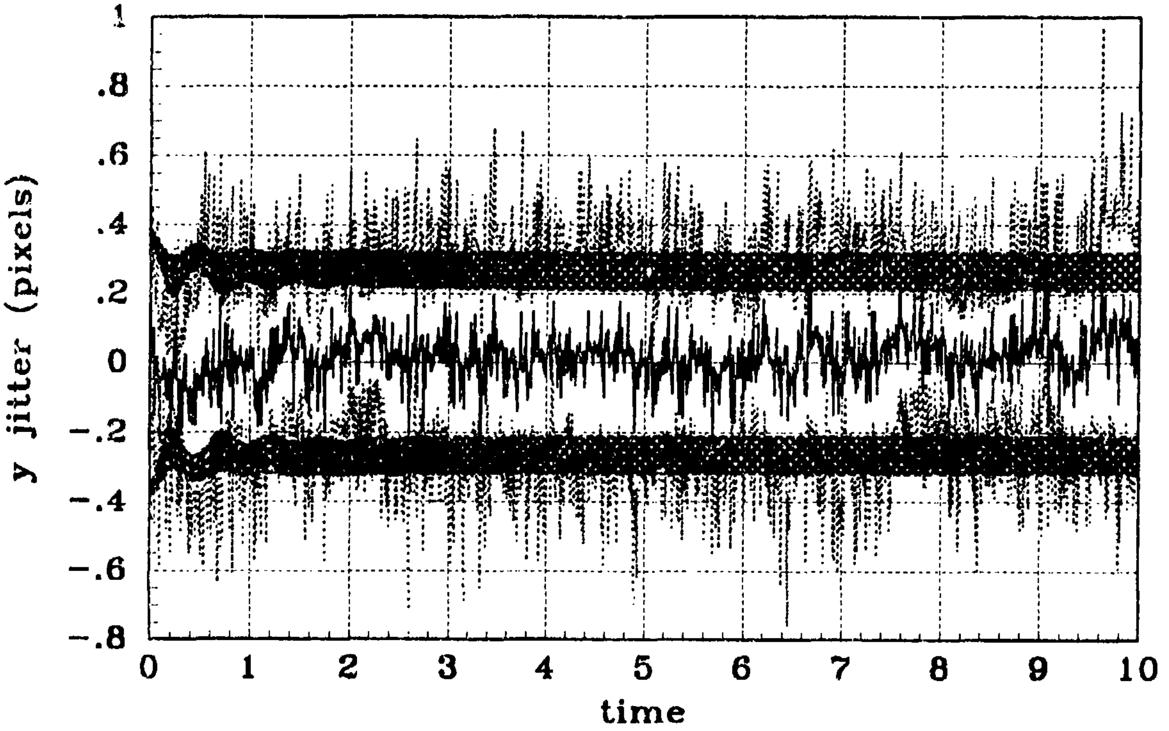


(b) X JITTER STATISTICS - Filter #10,Nonlinear, PC#4

Figure D.18 Pogo Condition #4, Simulation #10, General Angle, X-Atmospheric Jitter

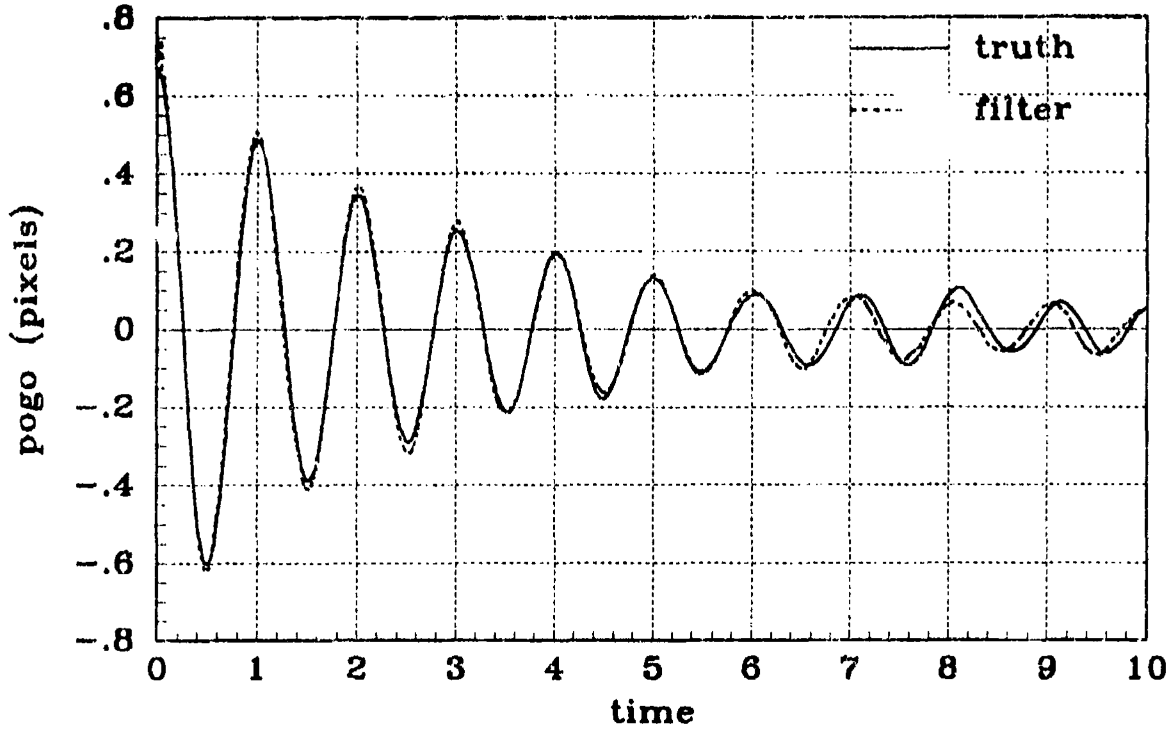


(a) Y JITTER - Filter #10,Nonlinear, PC#4

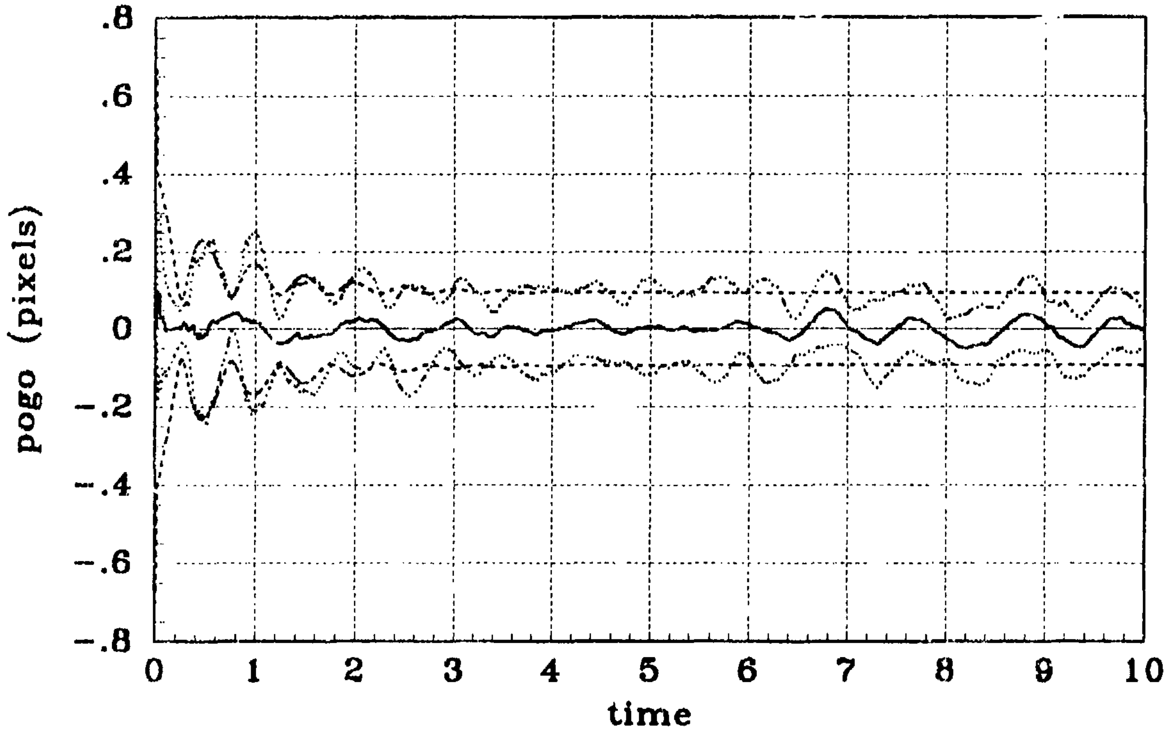


(b) Y JITTER STATISTICS - Filter #10,Nonlinear, PC#4

Figure D.19 Pogo Condition #4, Simulation #10, General Angle, Y-Atmospheric Jitter

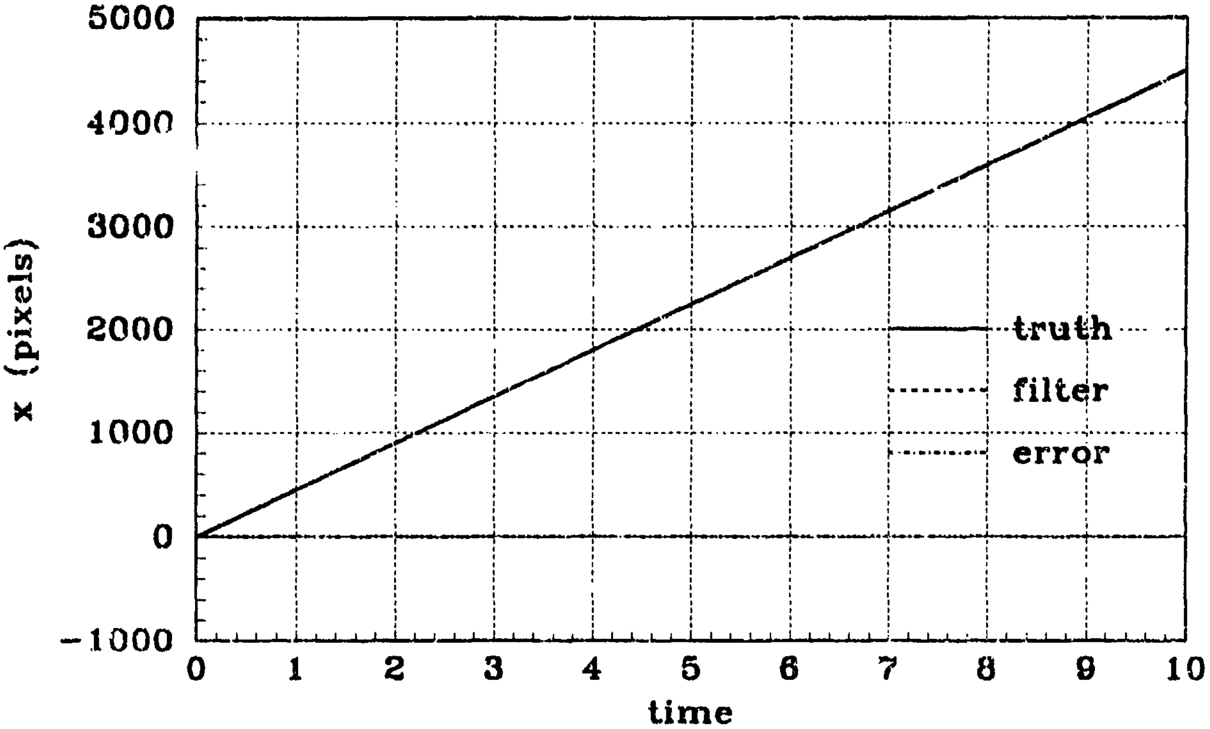


(a) POGO OFFSET − Filter #10,Nonlinear, PC#4

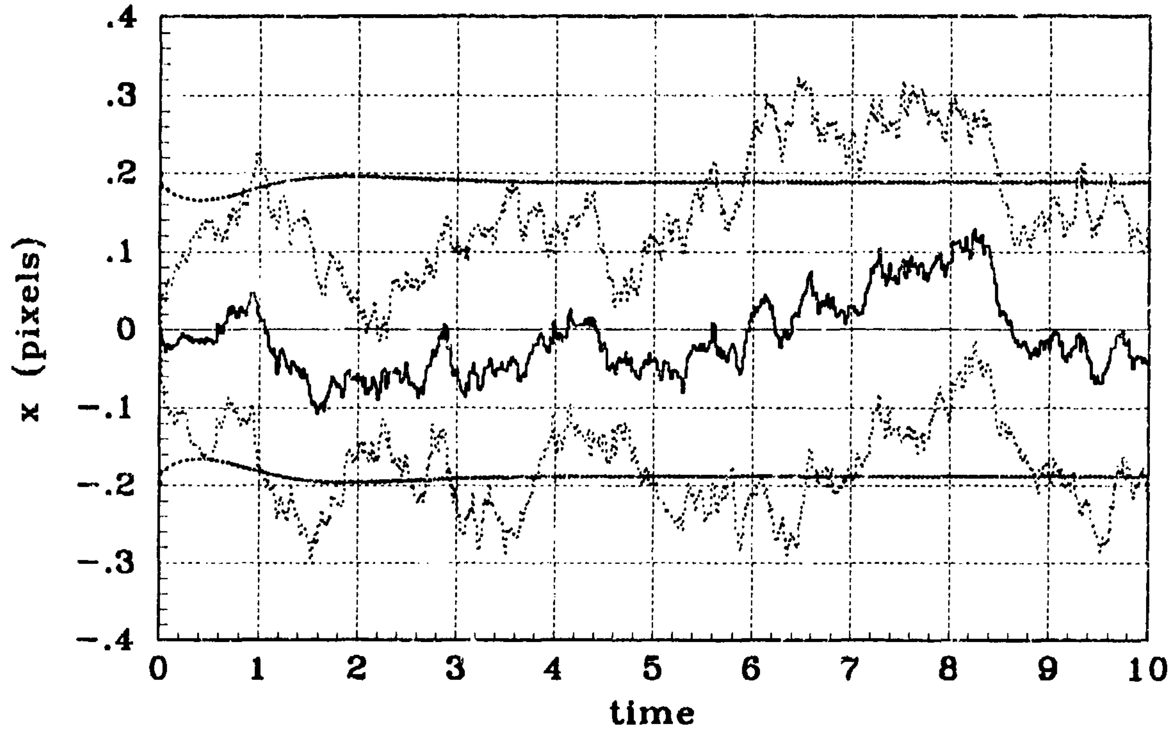


(b) POGO OFFSET STATISTICS − Filter #10,Nonlinear, PC#4

Figure D.20 Pogo Condition #4, Simulation #10, General Angle, Pogo Offset

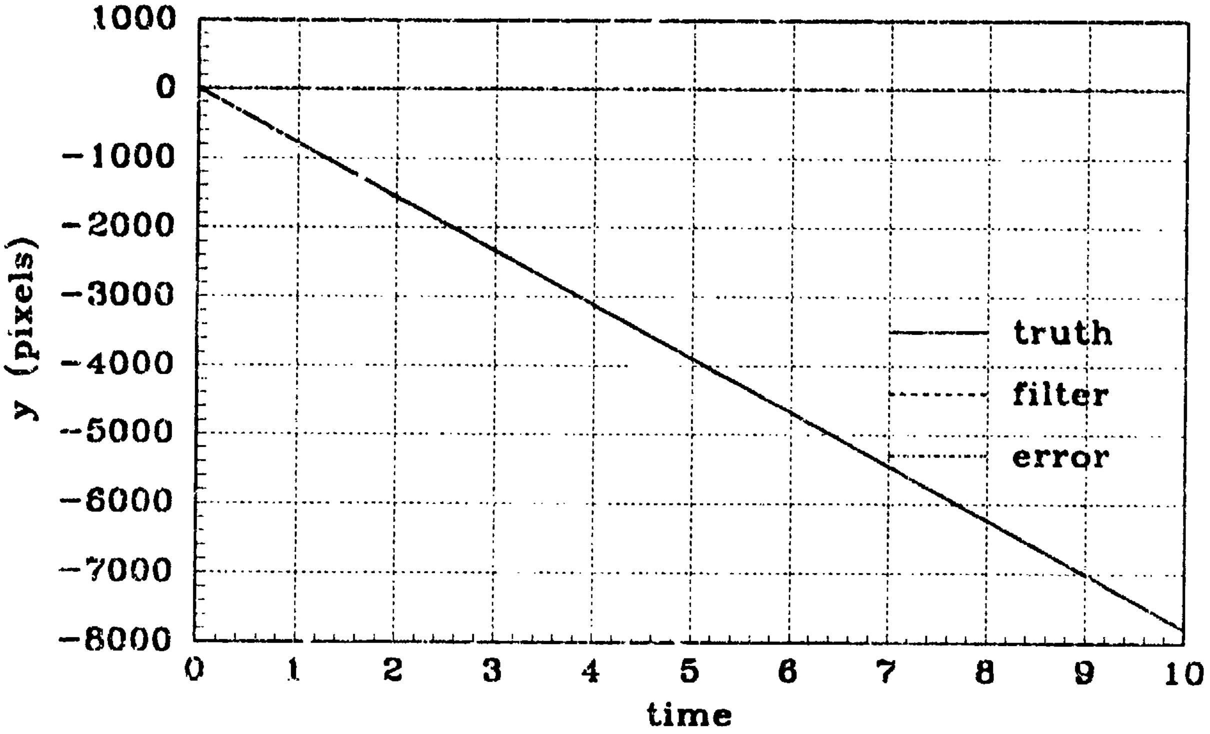


(a) X POSITION – Filter #10,Nonlinear, PC#5

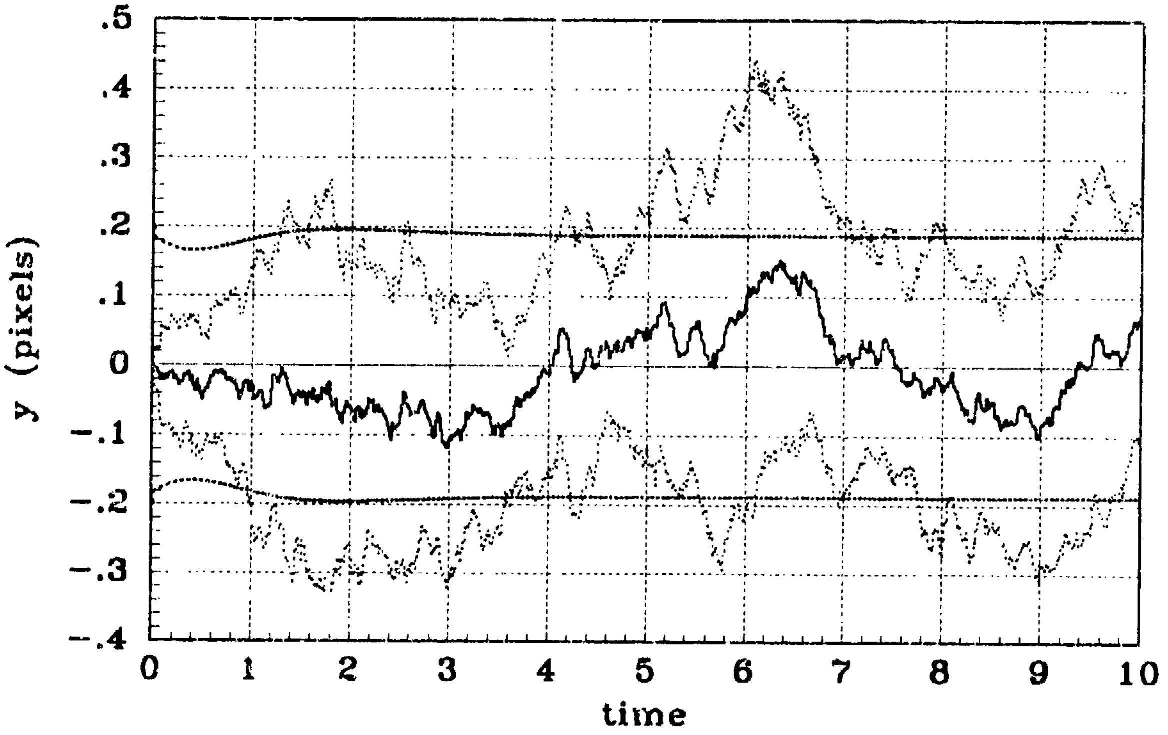


(b) X POSITION STATISTICS – Filter #10,Nonlinear, PC#5

Figure D.21 Pogo Condition #5, Simulation #10, General Angle, X-Position

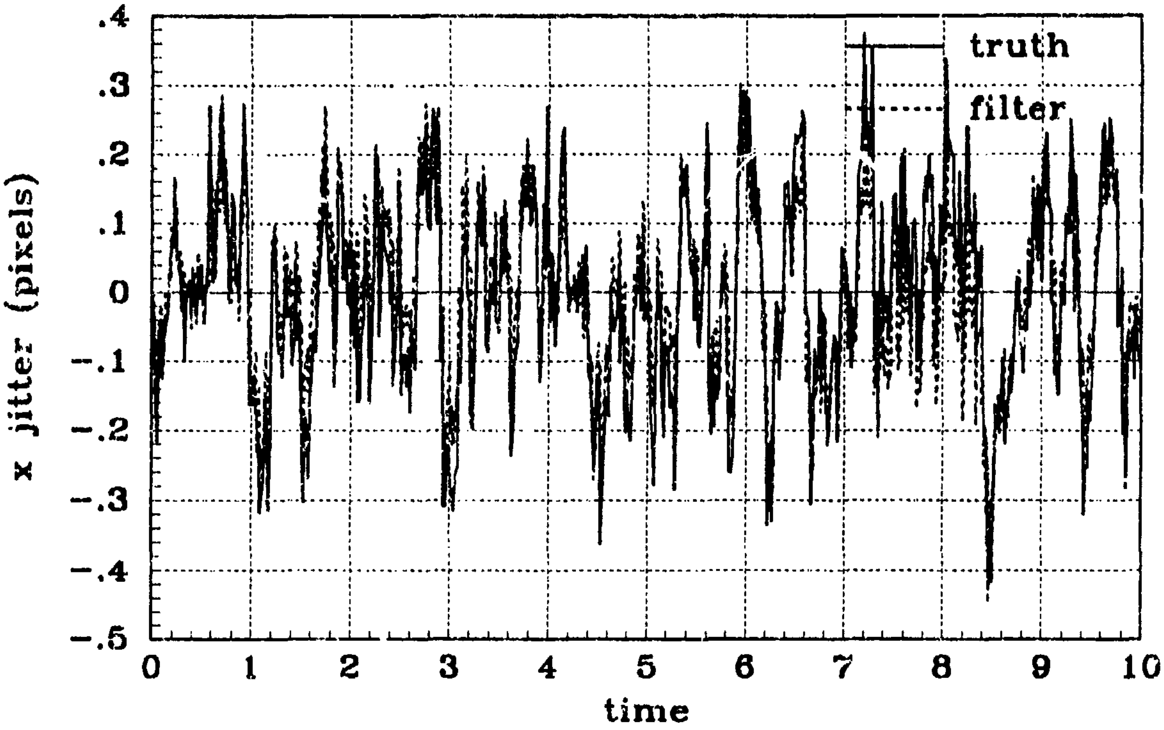


(a) Y POSITION - Filter #10,Nonlinear, PC#5

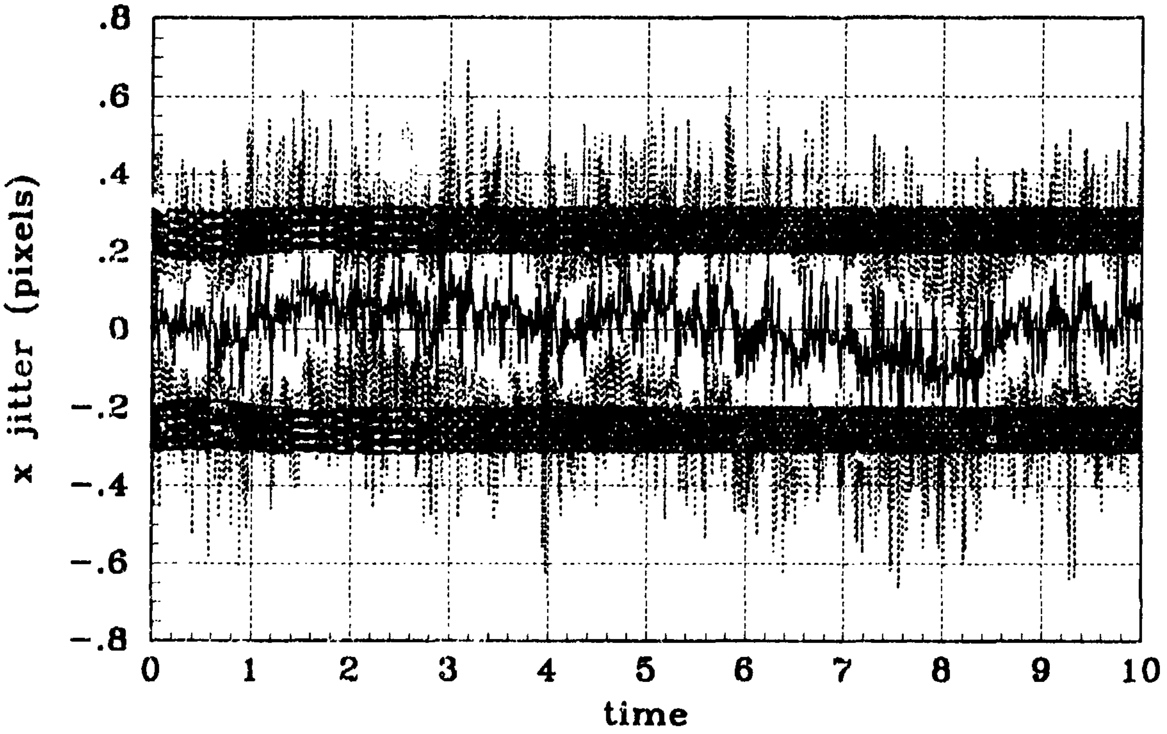


(b) Y POSITION STATISTICS - Filter #10,Nonlinear, PC#5

Figure D.22 Pogo Condition #5, Simulation #10, General Angle, Y-Position

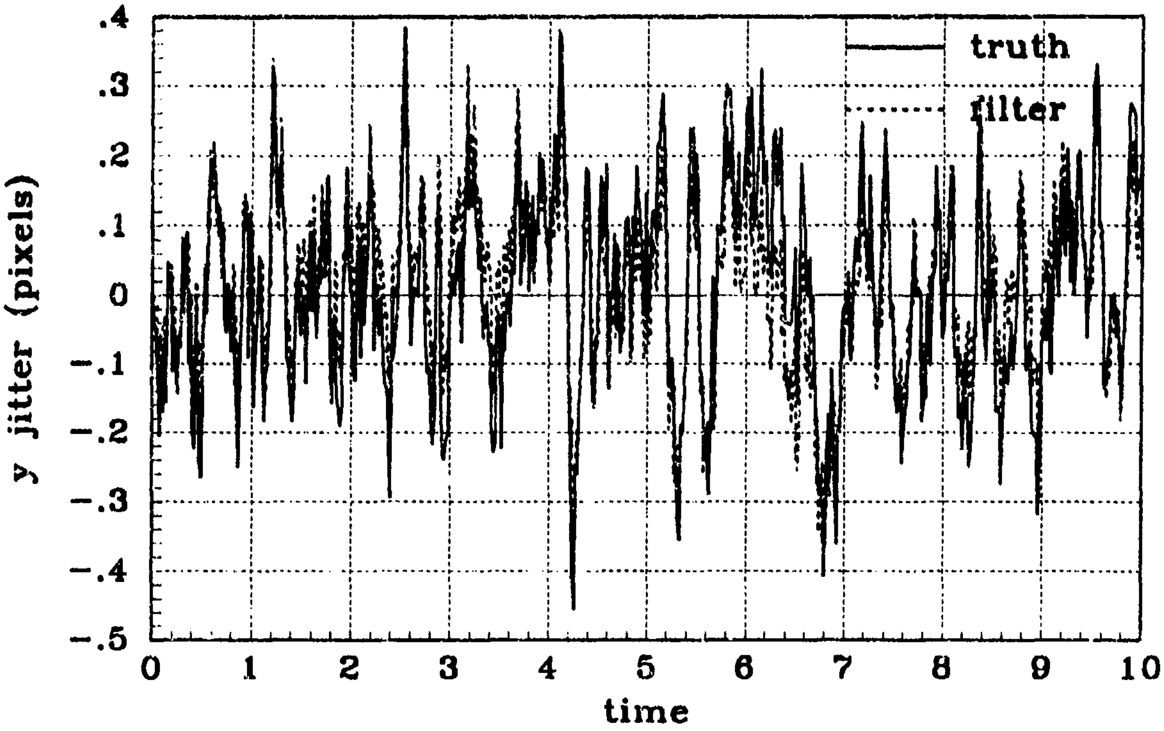


(a) X JITTER – Filter #10,Nonlinear, PC#5

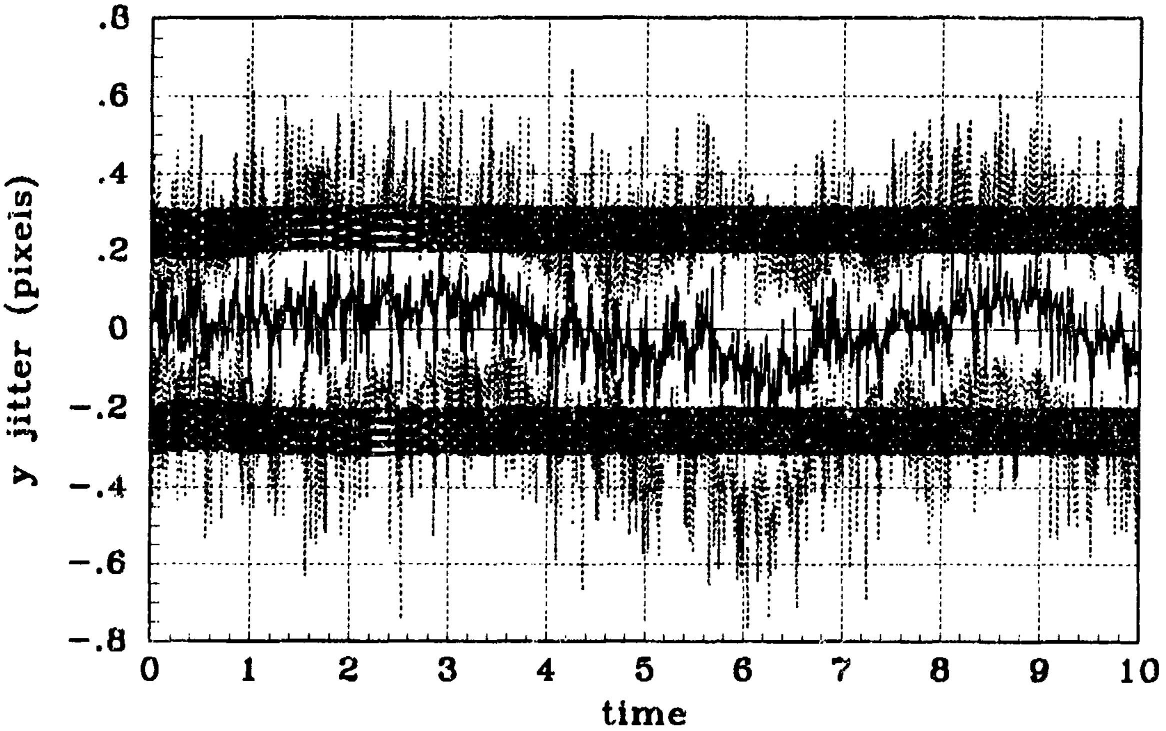


(b) X JITTER STATISTICS – Filter #10,Nonlinear, PC#5

Figure D.23 Pogo Condition #5, Simulation #10, General Angle, X-Atmospheric Jitter

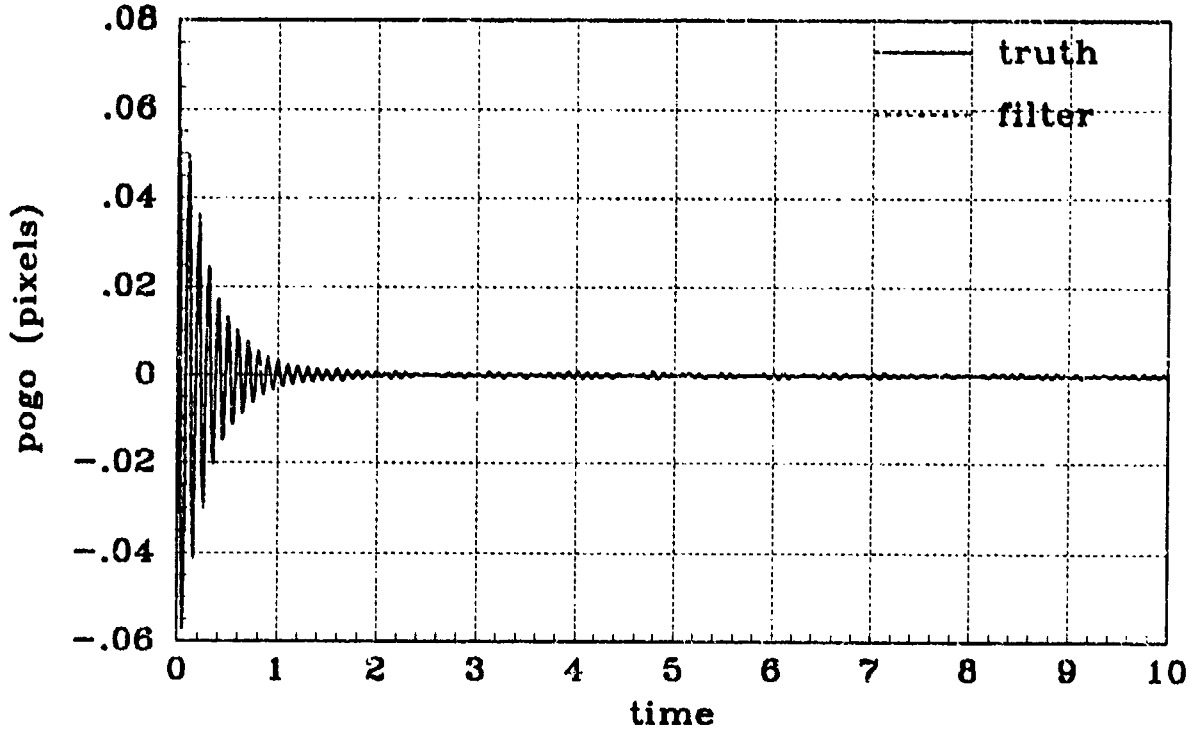


(a) Y JITTER - Filter #10,Nonlinear, PC#5

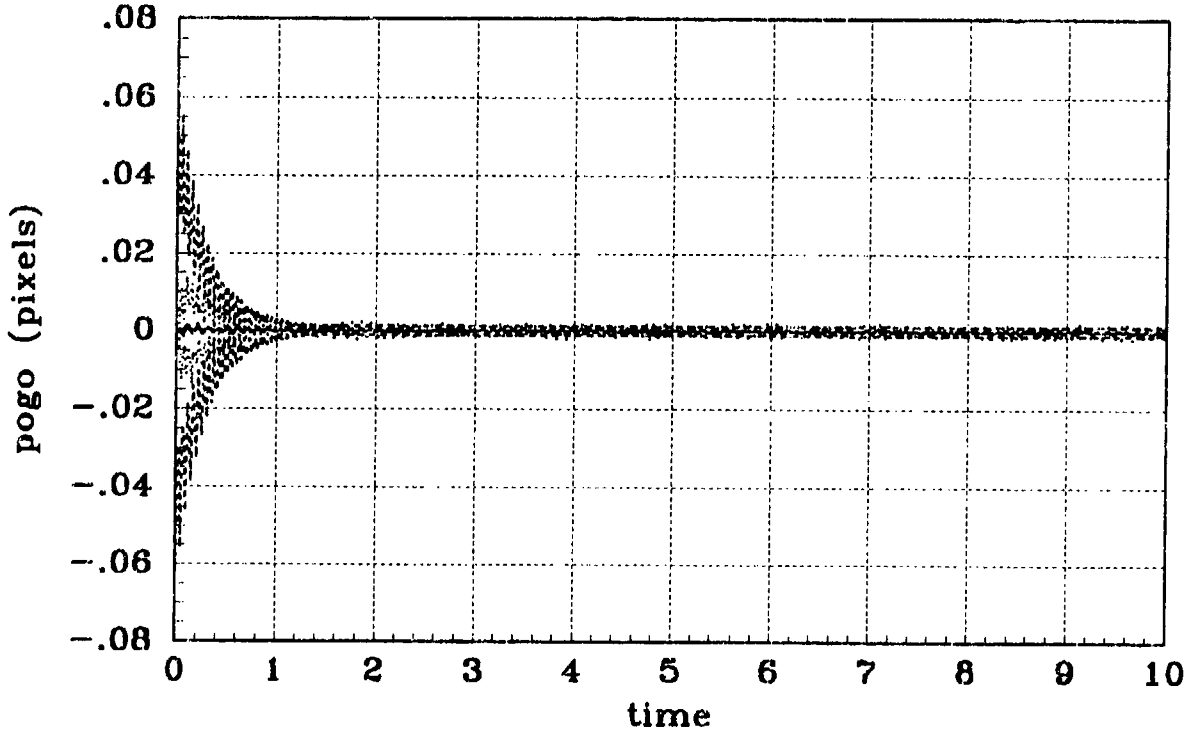


(b) Y JITTER STATISTICS - Filter #10,Nonlinear, PC#5

Figure D.24 Pogo Condition #5, Simulation #10, General Angle, Y-Atmospheric Jitter

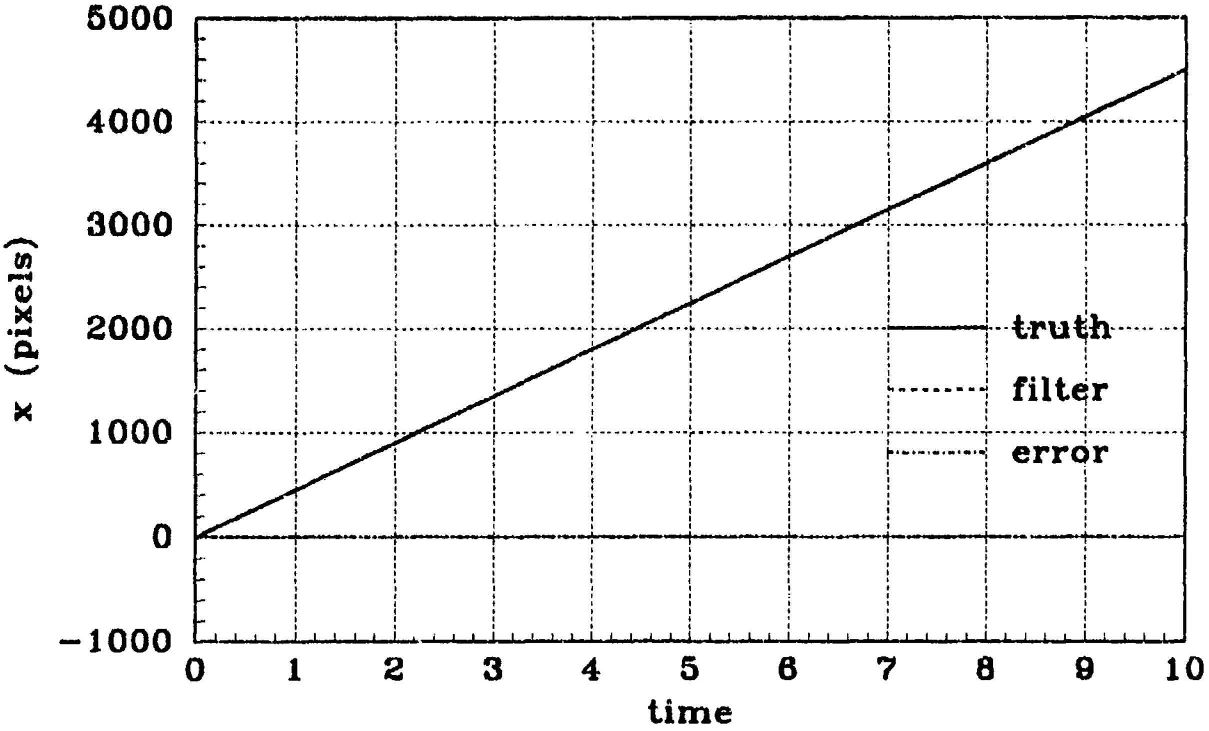


(a) POGO OFFSET – Filter #10,Nonlinear, PC#5

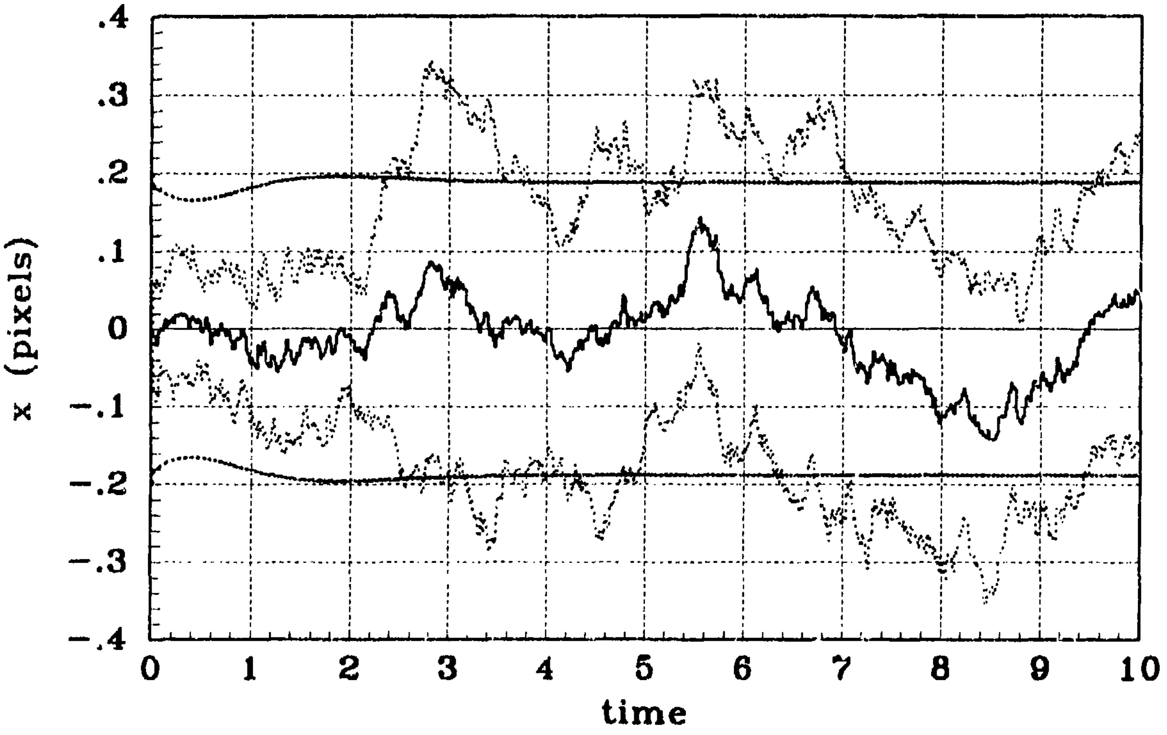


(b) POGO OFFSET STATISTICS – Filter #10,Nonlinear, PC#5

Figure D.25 Pogo Condition #5, Simulation #10, General Angle, Pogo Offset

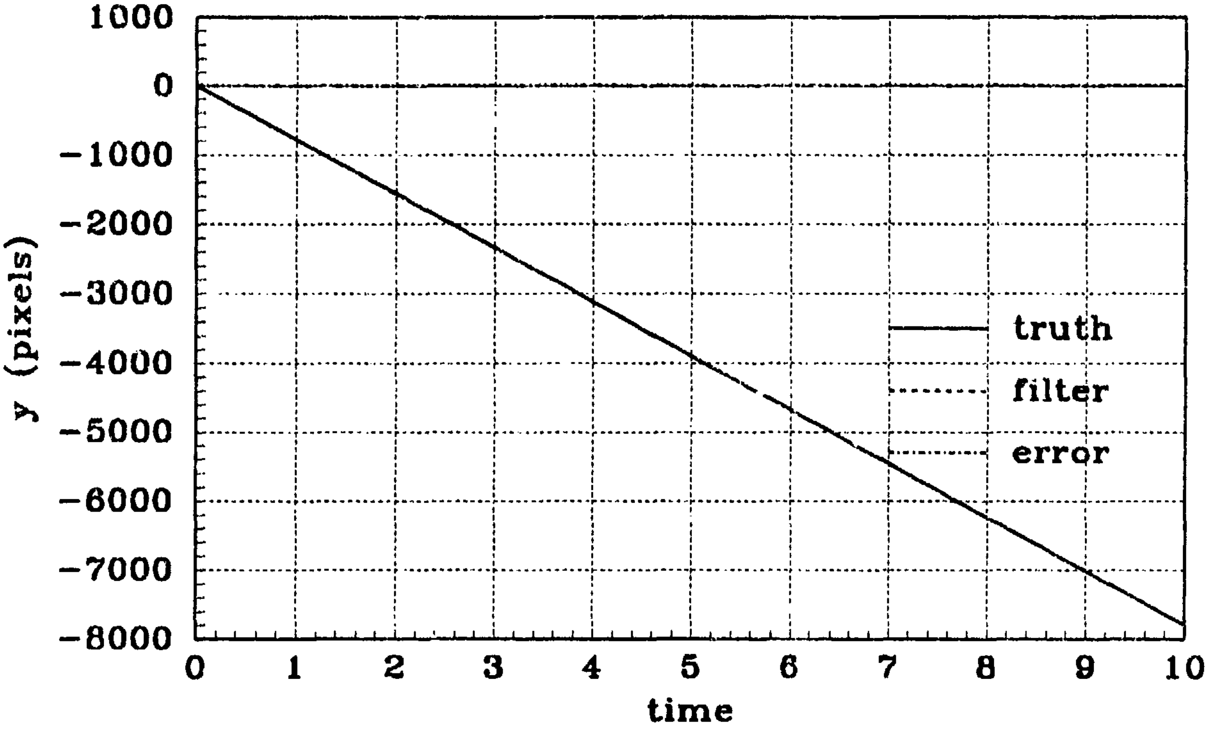


(a) X POSITION − Filter #10,Nonlinear, PC#6

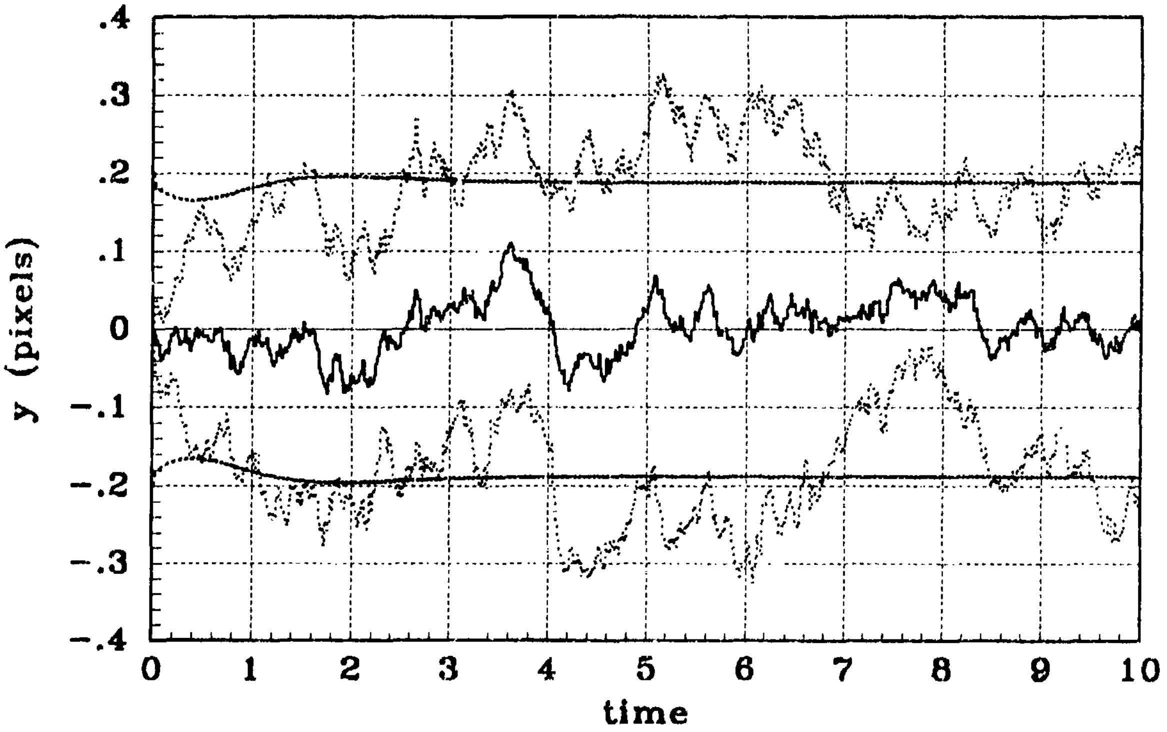


(b) X POSITION STATISTICS − Filter #10,Nonlinear, PC#6

Figure D.26 Pogo Condition #6, Simulation #10, General Angle, X-Position

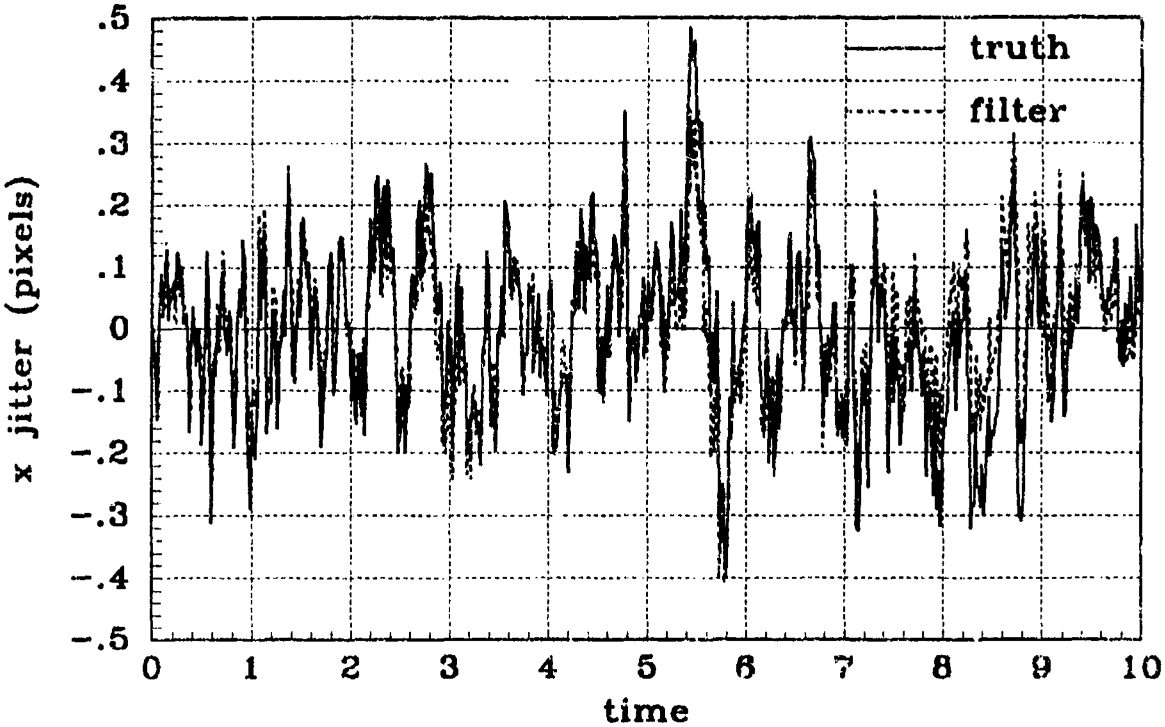


(a) Y POSITION – Filter #10,Nonlinear, PC#6

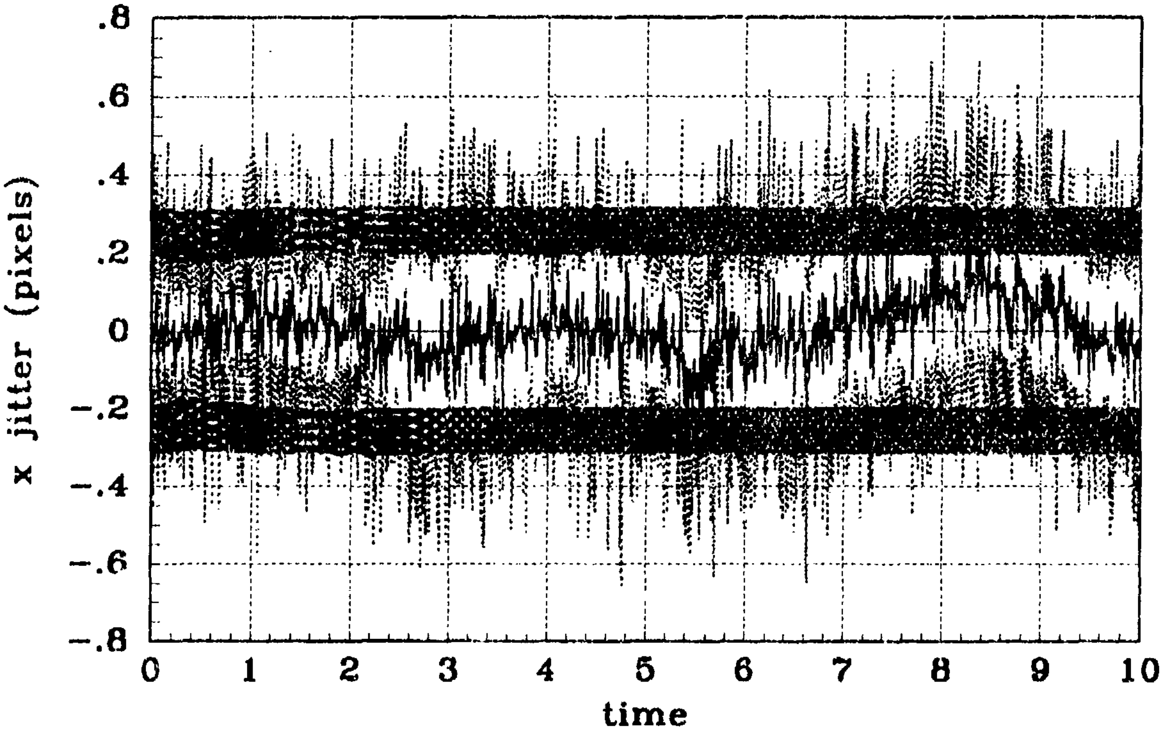


(b) Y POSITION STATISTICS – Filter #10,Nonlinear, PC#6

Figure D.27 Pogo Condition #6, Simulation #10, General Angle, Y-Position

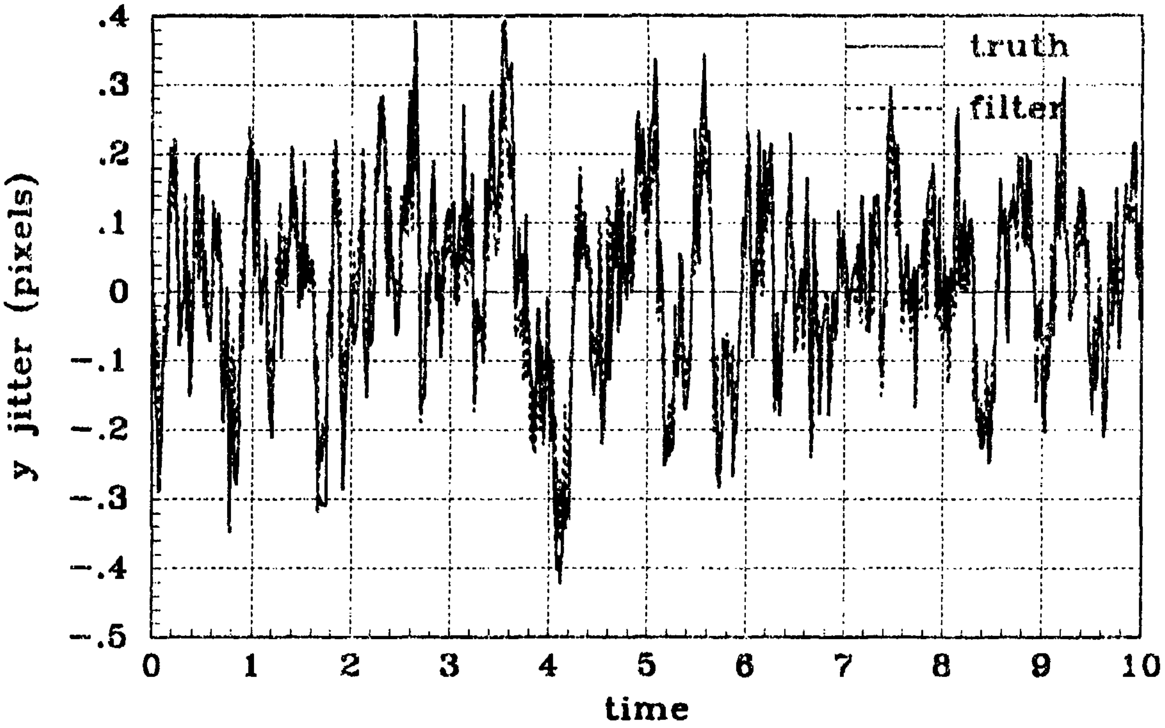


(a) X JITTER - Filter #10,Nonlinear, PC#6

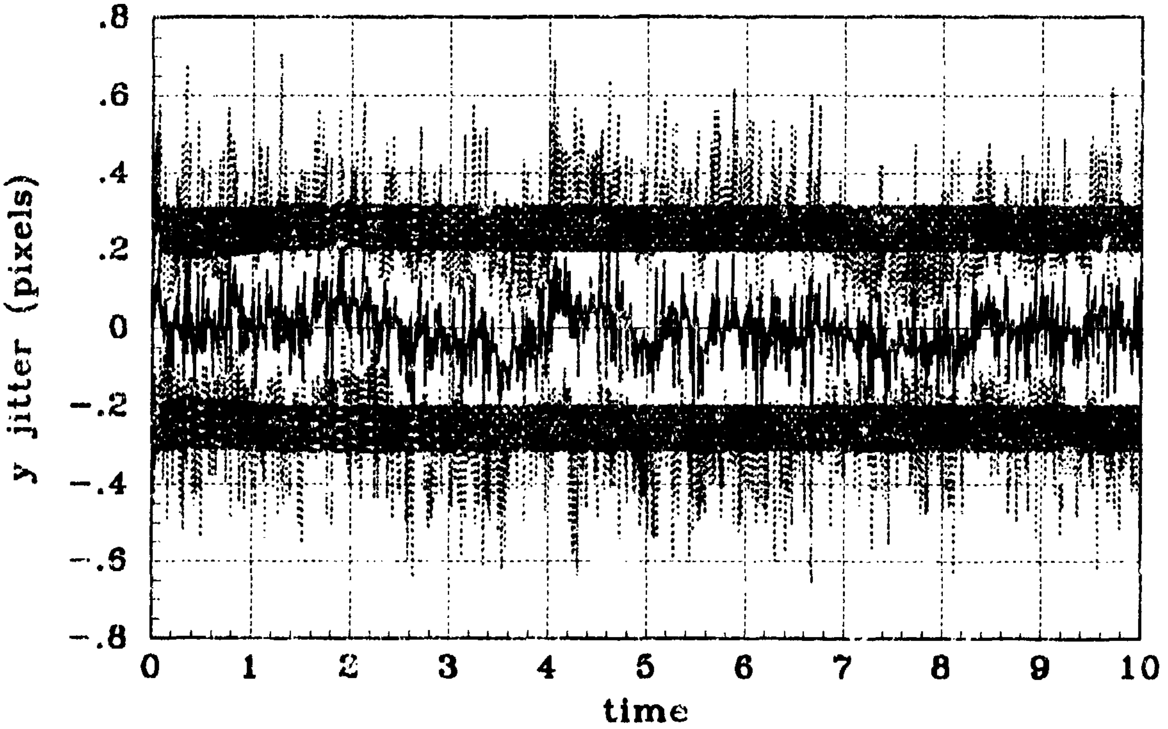


(b) X JITTER STATISTICS - Filter #10,Nonlinear, PC#6

Figure D.28 Pogo Condition #6, Simulation #10, General Angle, X-Atmospheric Jitter

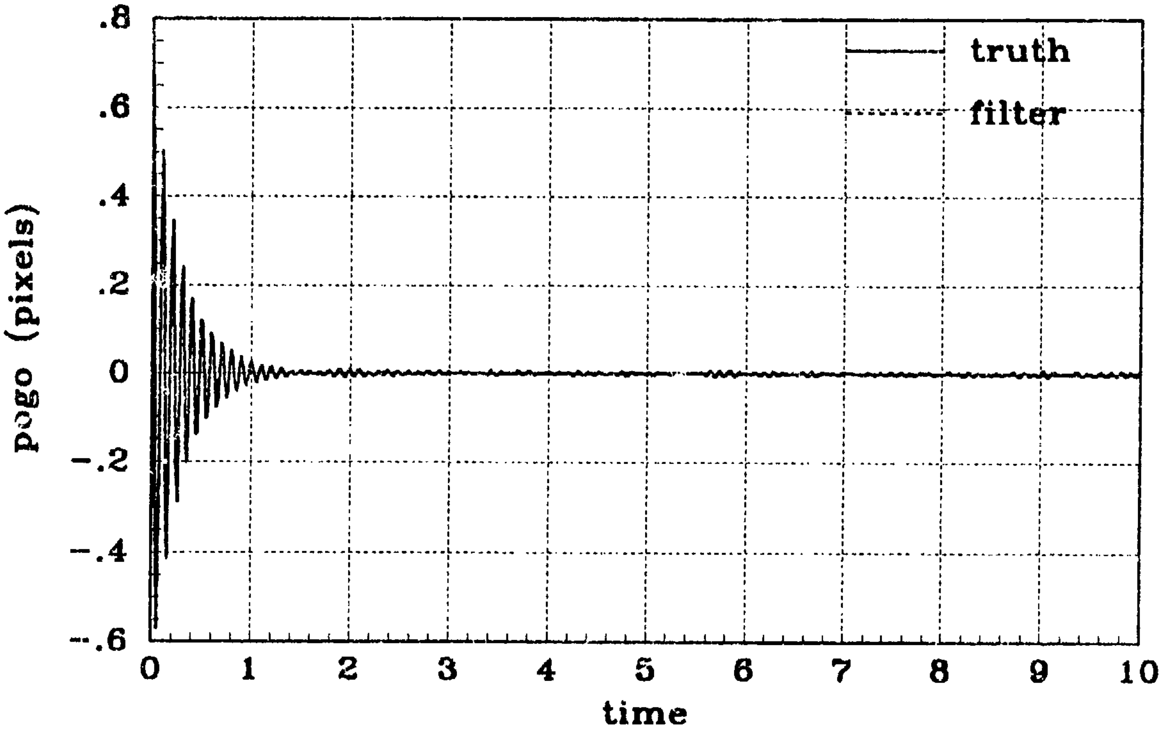


(a) Y JITTER – Filter #10,Nonlinear, PC#6

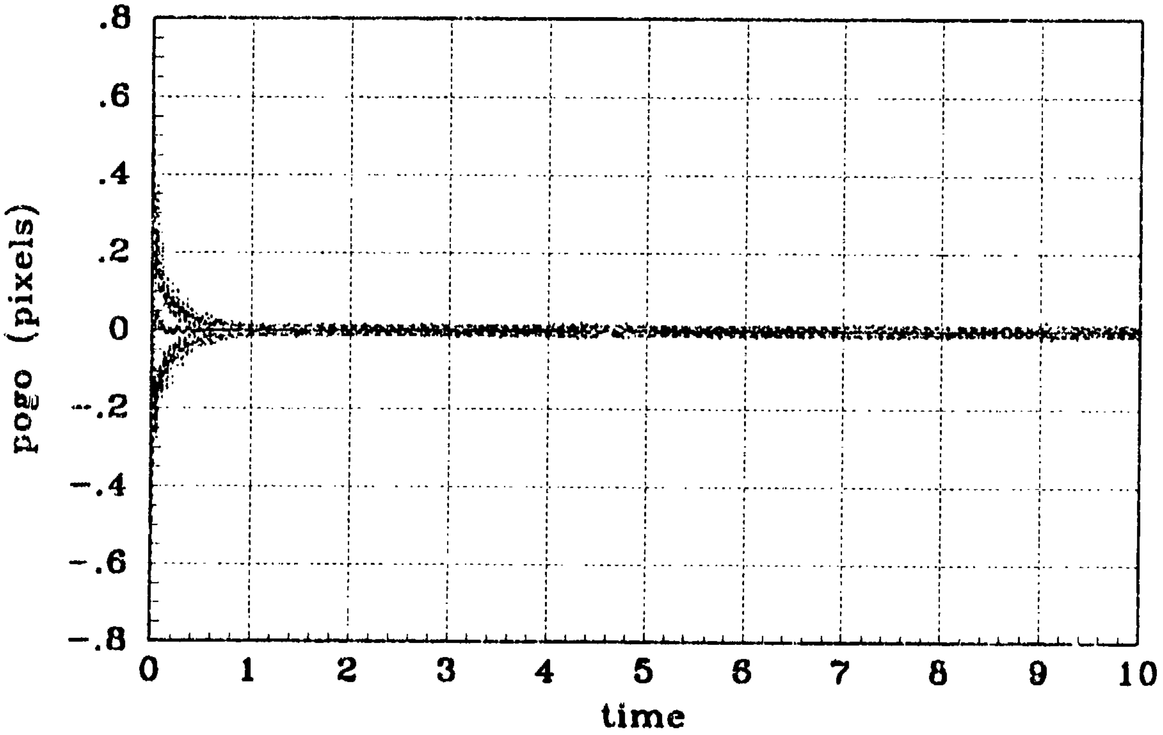


(b) Y JITTER STATISTICS – Filter #10,Nonlinear, PC#6

Figure D.29 Pogo Condition #6, Simulation #10, General Angle, Y-Atmospheric Jitter



(a) POGO OFFSET – Filter #10,Nonlinear, PC#6



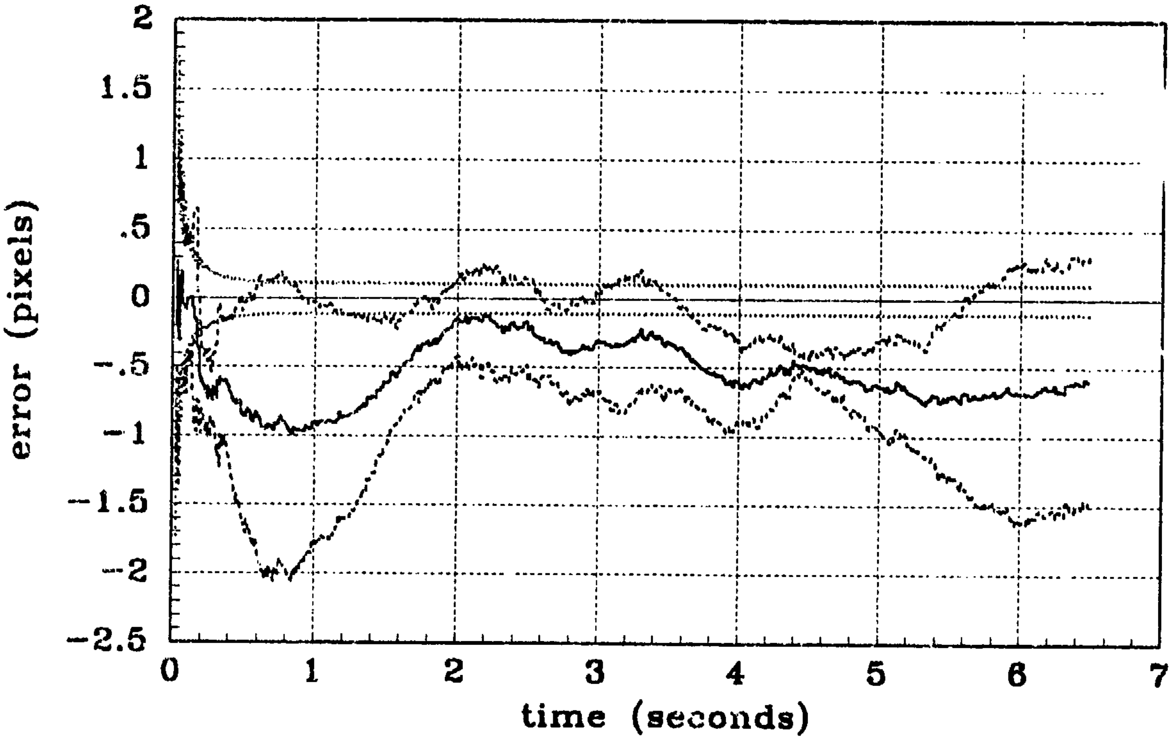
(b) POGO OFFSET STATISTICS – Filter #10,Nonlinear, PC#6

Figure D.30 Pogo Condition #6, Simulation #10, General Angle, Pogo Offset

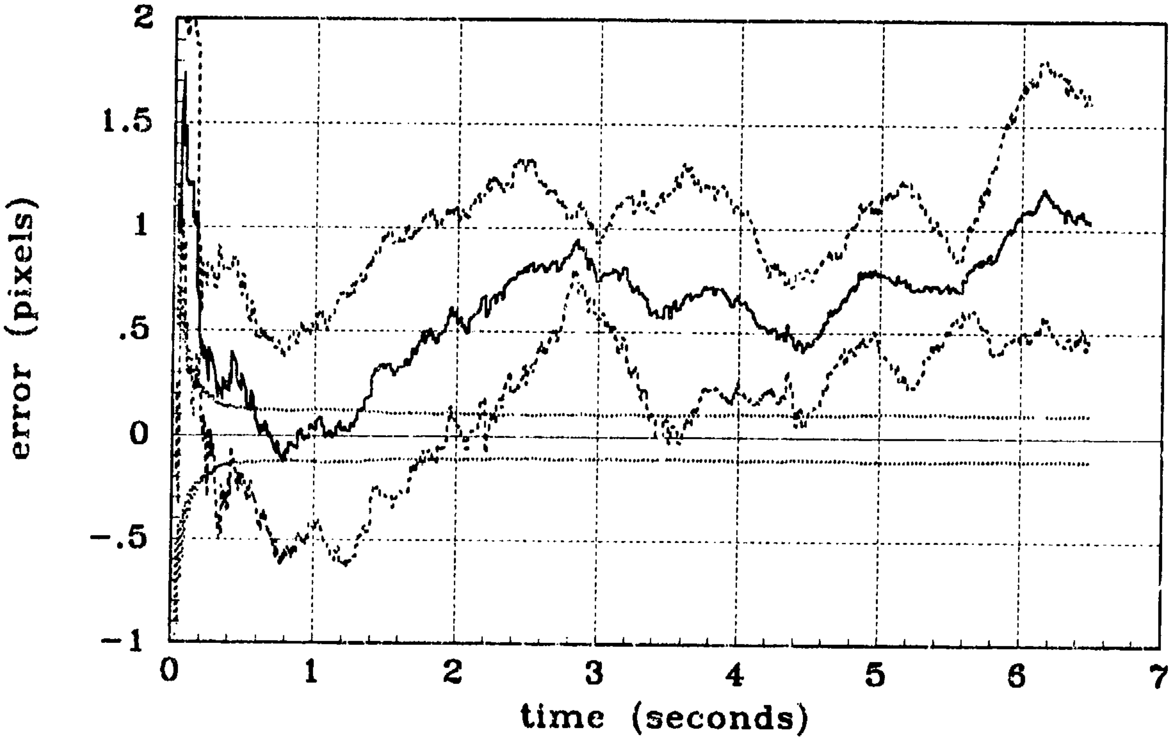
Appendix E

AFIT Elemental Filter Performance Plots

This appendix contains the average state and error statistics plots of the nine-state AFIT elemental filter. The data depicted in the two types of plots in this appendix are explained in Appendices A and B. The state comparison plots show the ensemble average truth state over the 5 Monte Carlo runs compared to the same statistic for the filter estimate. The error statistics plots represent the error mean and mean ±1 standard deviation values in pixels (or pixels/second for velocity and pogo velocity), of the errors between the filter estimated and true state.

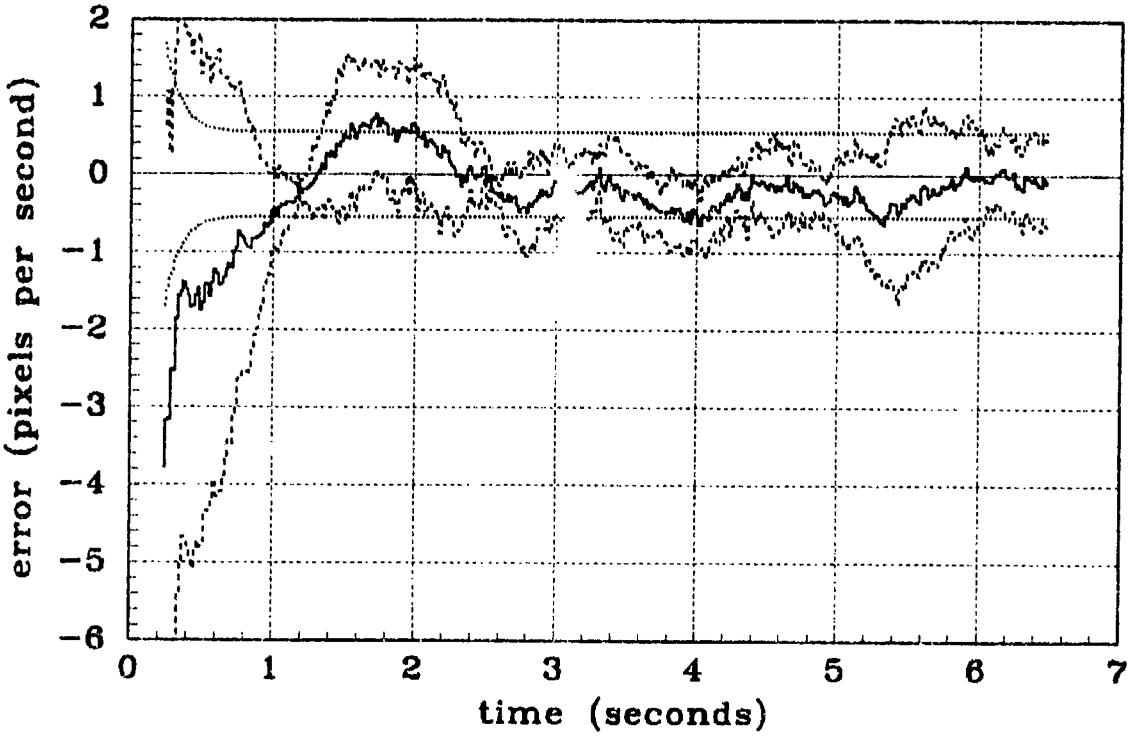


(a) FILTER vs ACTUAL ERROR (X-POSITION)

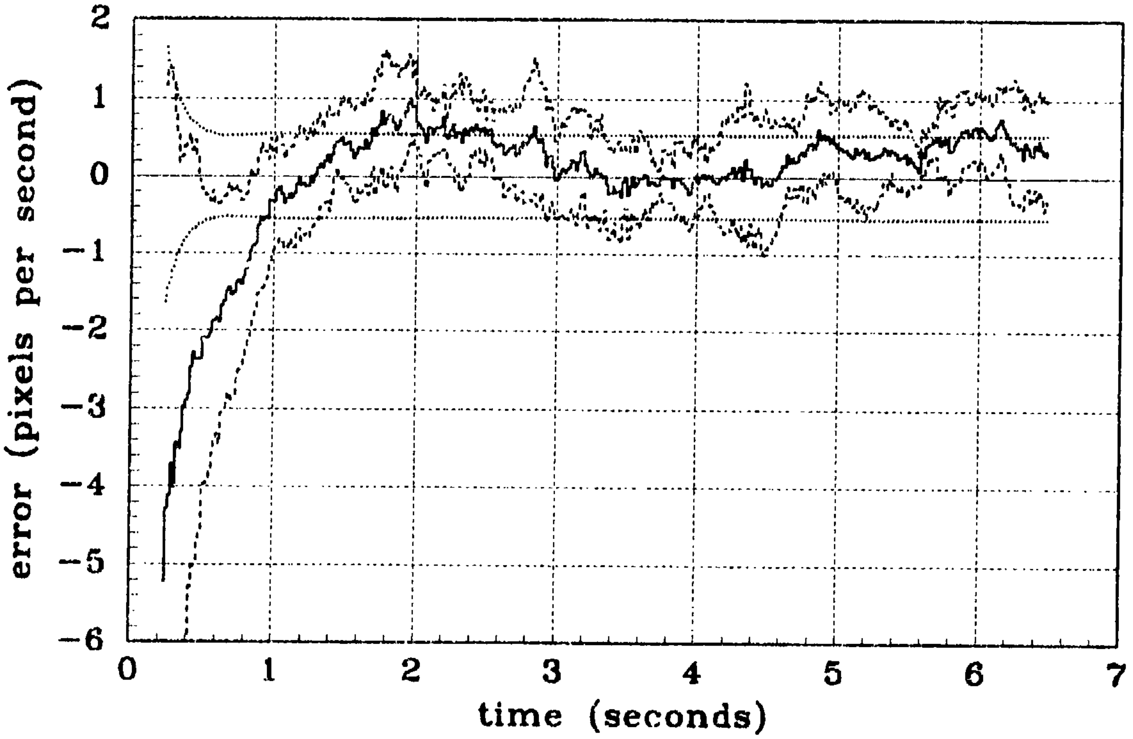


(b) FILTER vs ACTUAL ERROR (Y-POSITION)

Figure E 1 X/Y Position Error Statistics

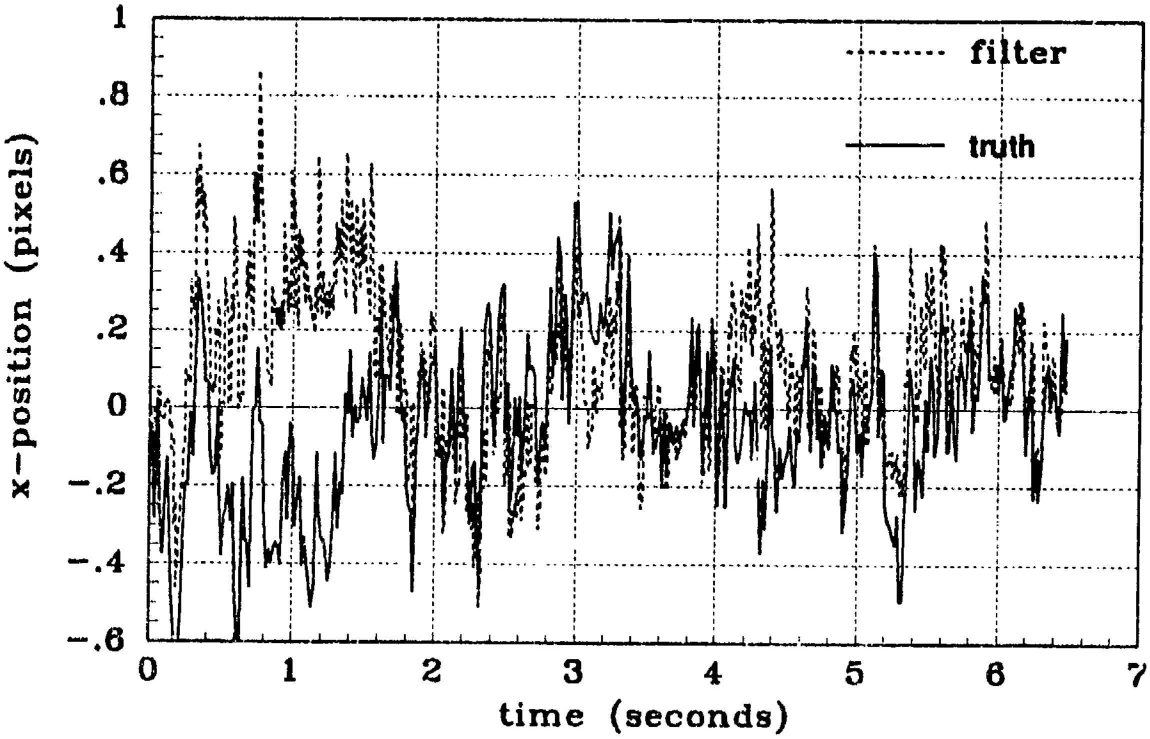


(a) FILTER vs ACTUAL ERROR (X−VELOCITY)

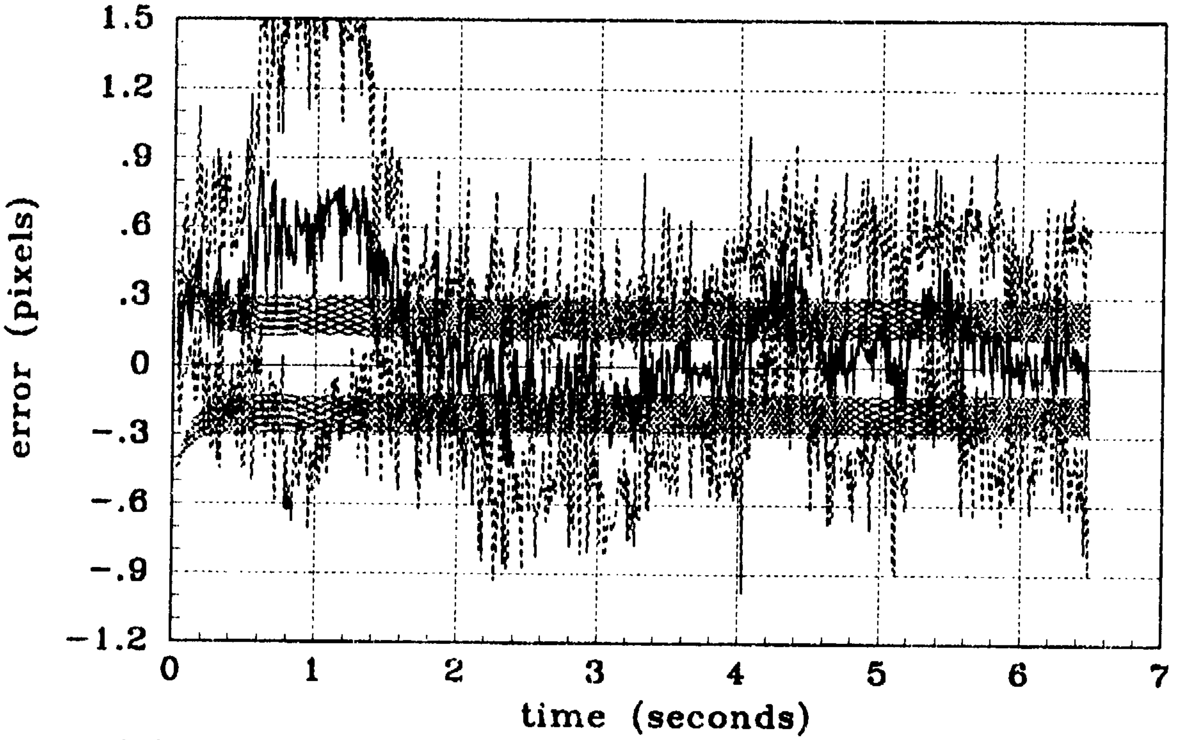


(b) FILTER vs ACTUAL ERROR (Y−VELOCITY)

Figure E.2 X/Y Velocity Error Statistics

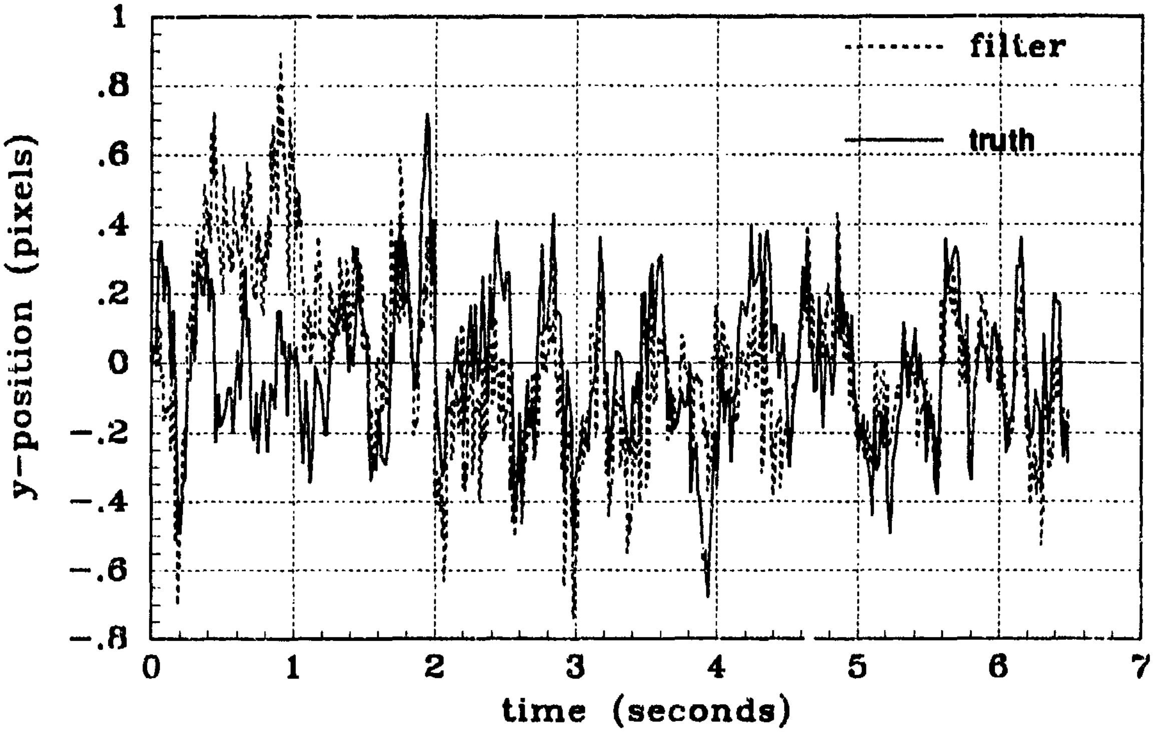


(a) FILTER vs TRUE X−ATMOSPHERE

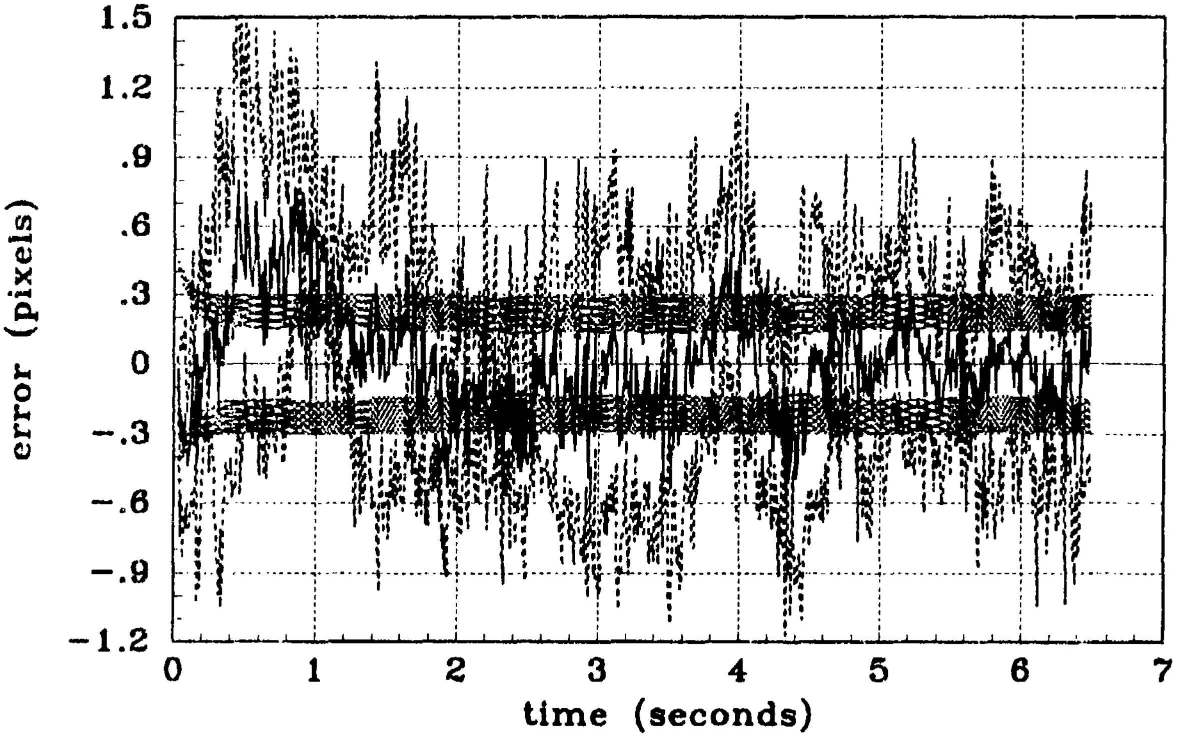


(b) FILTER vs ACTUAL ERROR (X−ATMOSPHERE)

Figure E.3 X Atmospheric Jitter

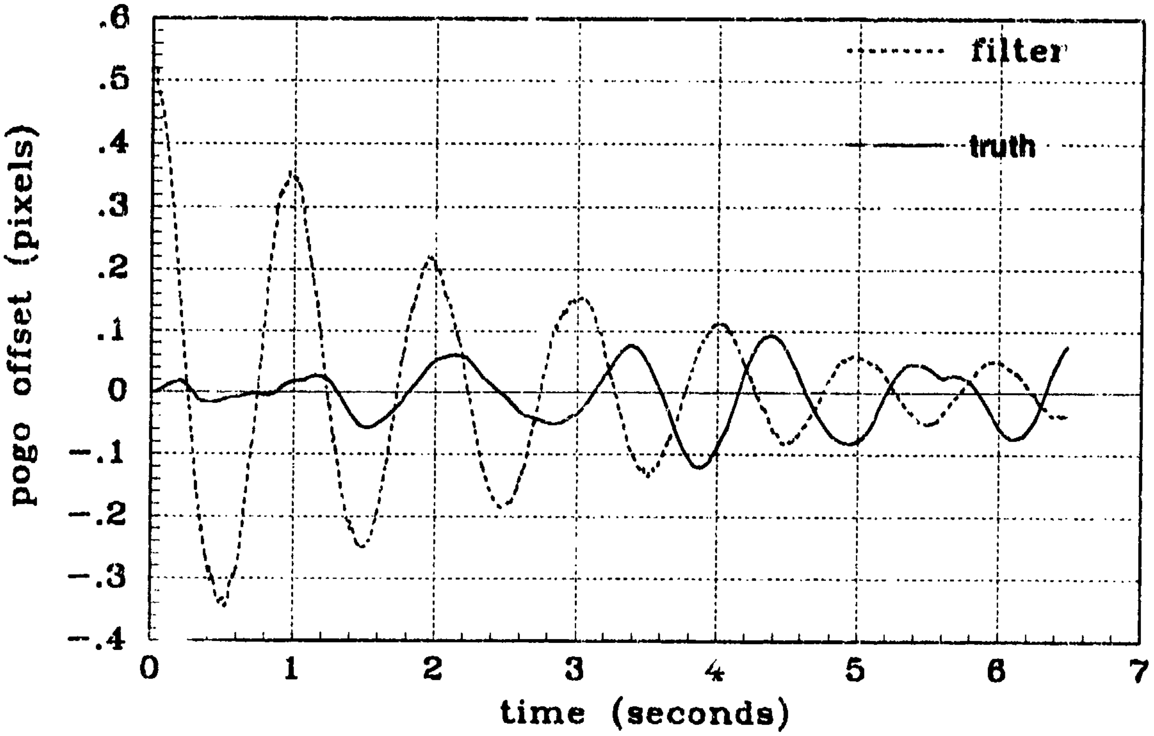


(a) FILTER vs TRUE Y−ATMOSPHERE

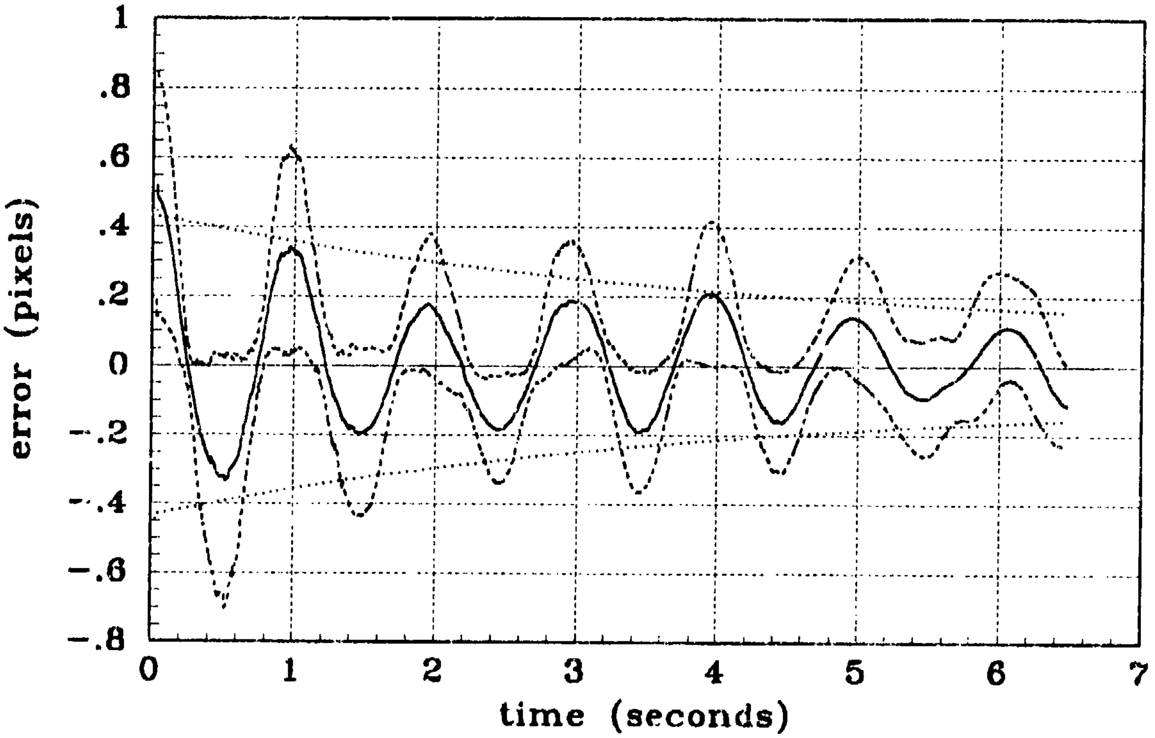


(b) FILTER vs ACTUAL ERROR (Y−ATMOSPHERE)

Figure E.4 Y Atmospheric Jitter

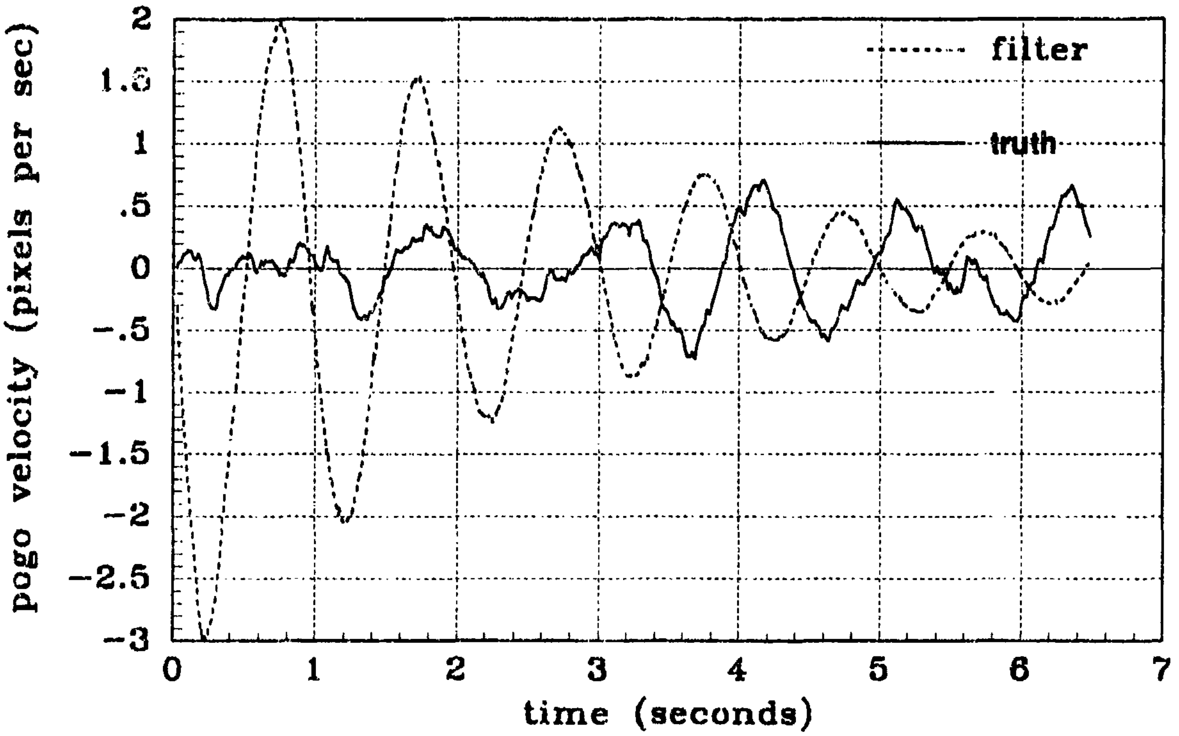


(a) FILTER vs TRUE POGO

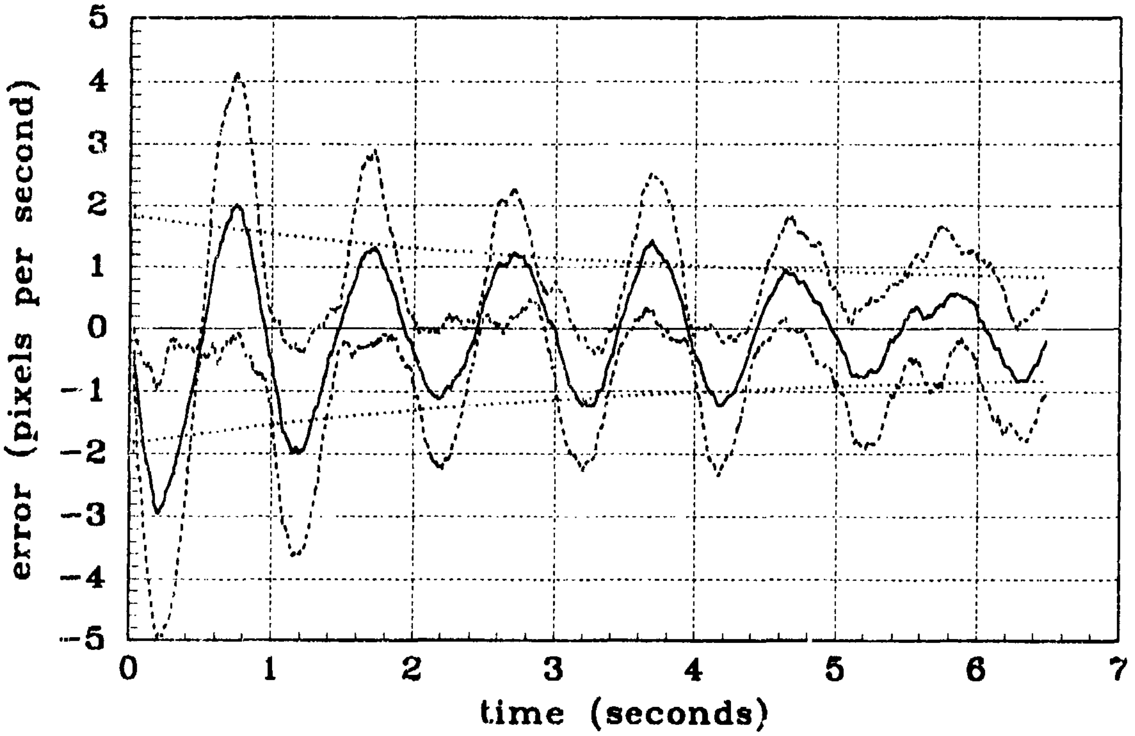


(b) FILTER vs ACTUAL ERROR (POGO POSITION)

Figure E.5 Pogo Offset

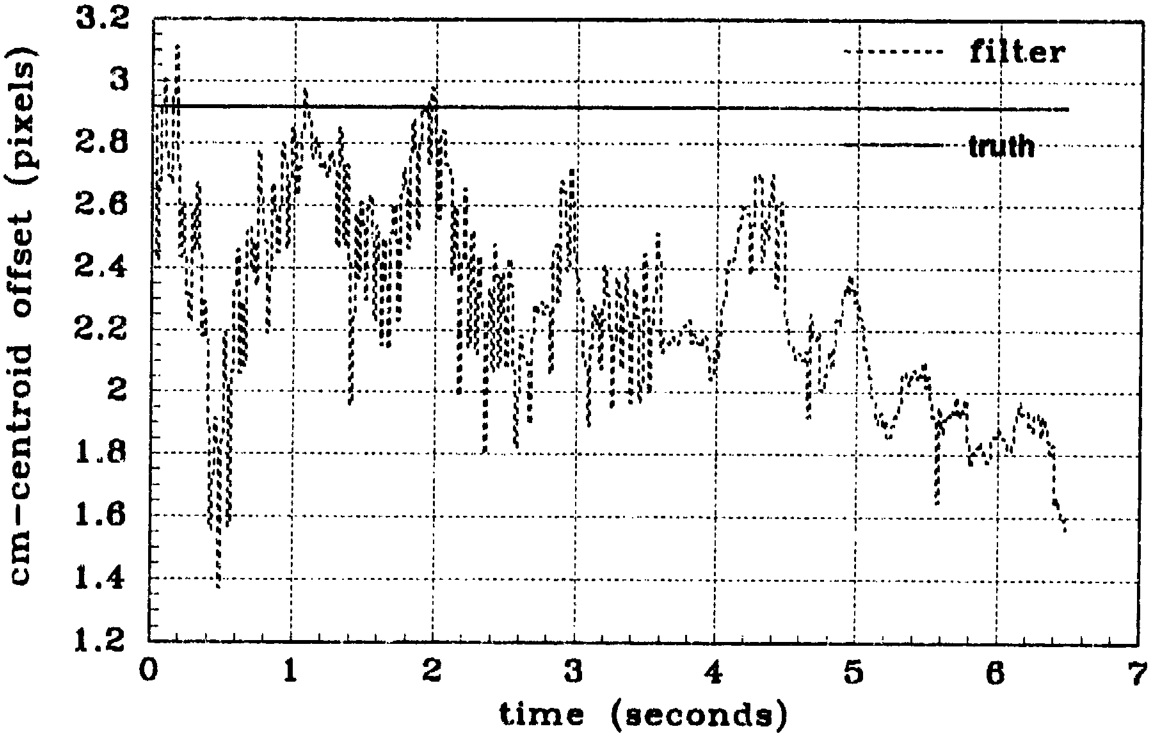


(a) FILTER vs TRUE POGO VELOCITY

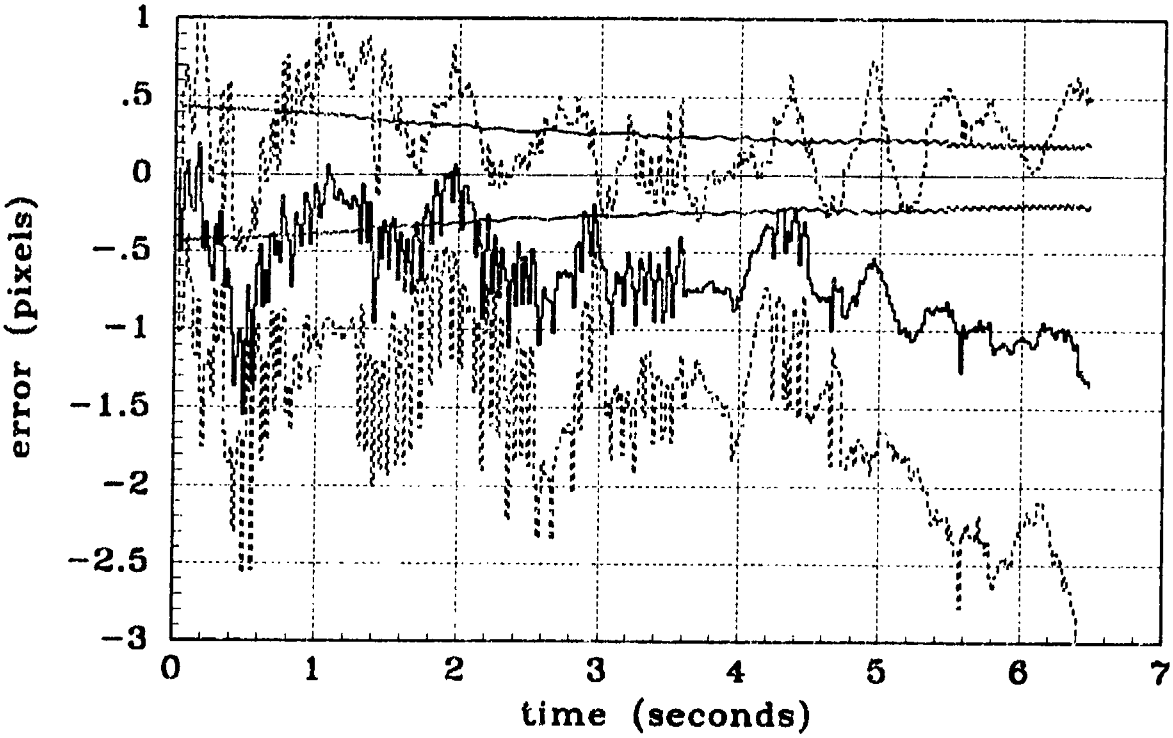


(b) FILTER vs ACTUAL ERROR (POGO VELOCITY)

Figure E.6 Pogo Velocity

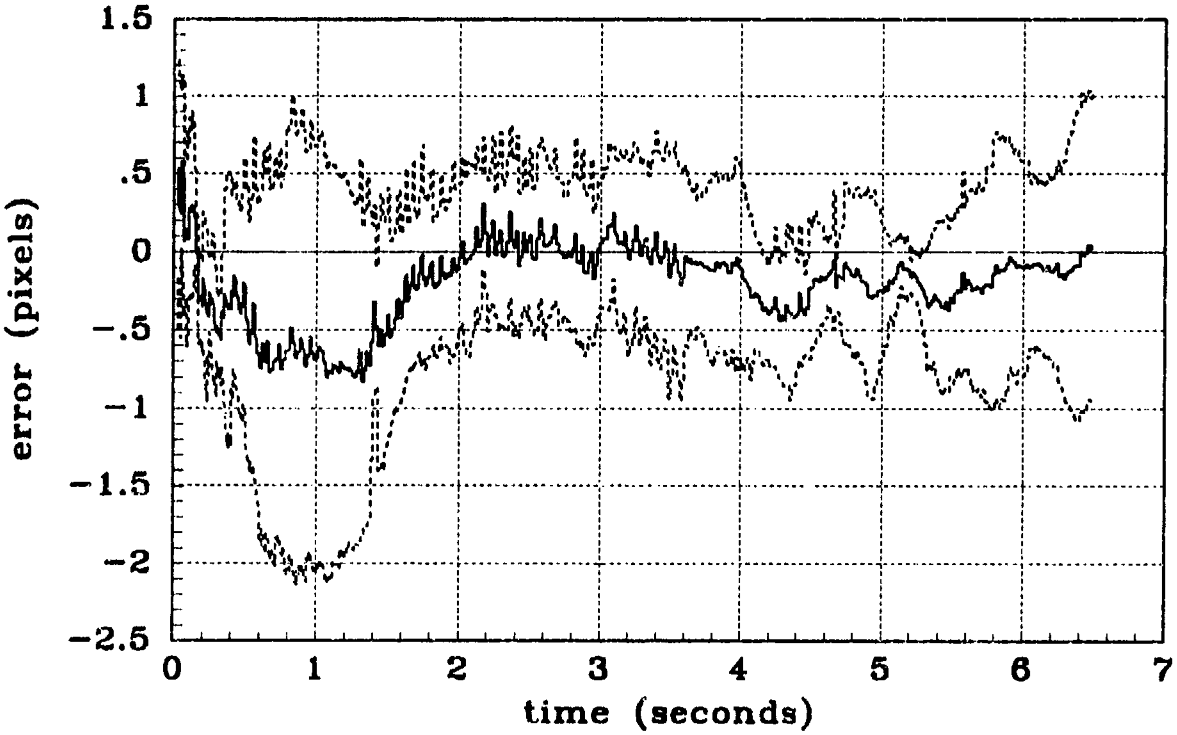


(a) FILTER vs TRUE COM-EQUILIBRIUM POINT OFFSET

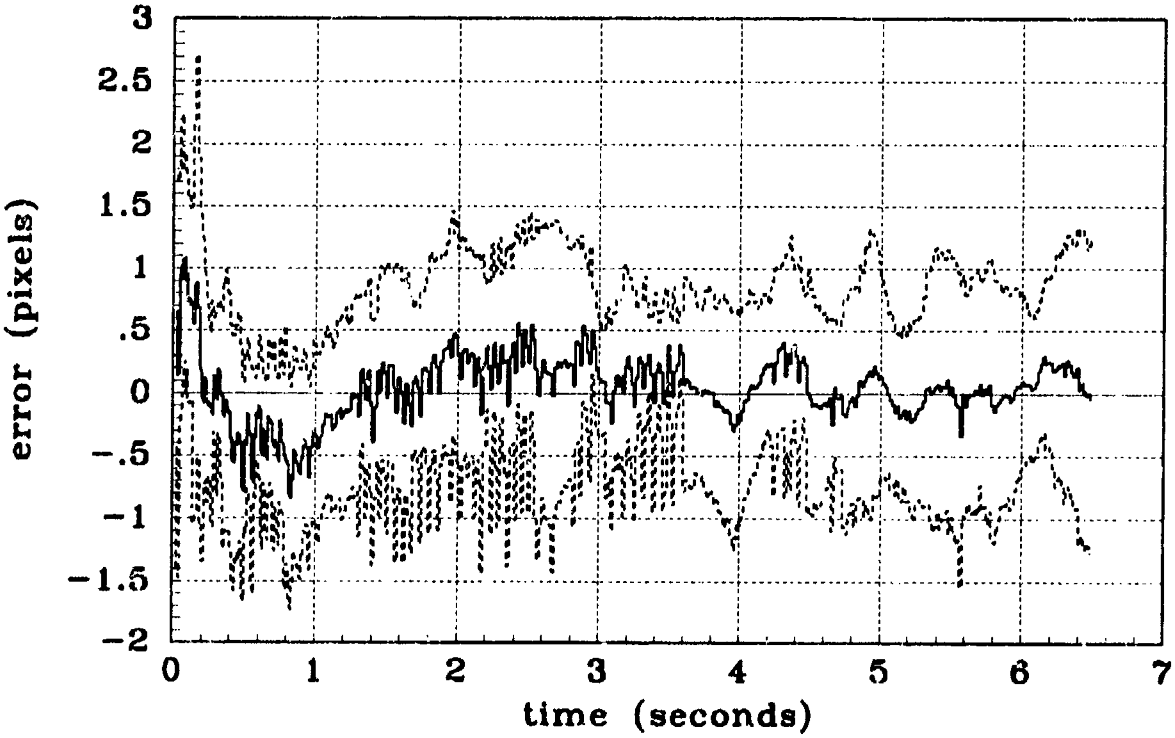


(b) FILTER vs ACTUAL ERROR (COM−EQUILIBRIUM POINT OFFSET)

Figure E.7 Center-of-Mass to Equilibrium Point Offset



(a) FILTER vs ACTUAL ERROR (X-CENTROID POSITION)



(b) FILTER vs ACTUAL ERROR (Y-CENTROID POSITION)

Figure E.8 Plume Centroid Error Statistics

Bibliography

1. "Accords on SDI, Cruise Missiles, Mobile ICBMs Aid START Process," *Aviation Week and Space Technology*, Vol 131, 25, October 2, 1989.

2. Bergin, J., Capt. Phillips Laboratory, Kirtland AFB, NM. Letter. 18 March 1991.

3. Bergin, J., Capt. Personal Interview. Phillips Laboratory, Kirtland AFB, NM, 18 - 19 June 1992.

4 Bergin, J., Capt. *High Altitude Balloon Experiment (HABE) Preliminary Design Review.* Slide presentation. Phillips Laboratory, Kirtland AFB, NM. 18 May, 1992.

5. Bergin, J., Capt. *HABE Release 1.2 Target Trajectories.* Working files from the Analytic Sciences Corporation. Phillips Laboratory, Kirtland AFB, NM. 11 May 1992.

6. Carlson, Neal, and Stanton Musick. *User's Manual for a Multimode Simulation for Optimal Filter Evaluation (MSOFE).* AFWAL-TR-88-1138, Wright-Patterson AFB OH: Avionics Lab, AFWAL/AARN-2, April 1990.

7. Durst, F., A. Melling, and J. H. Whitelaw. *Principles and Practice of Laser-Doppler Anemometry.* New York: Academic Press, 1981.

8. Eden, Claude W. *Enhanced Tracking of Ballistic Targets Using Forward Looking Infrared Measurements With Active Target Illumination.* MS Thesis, AFIT/GE/ENG/89D-11. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH December 1989.

9. Evans, Roger J. *Kalman Filter Tracking of a Reflective Target Using Forward Looking Infrared Measurements and Laser Illumination.* MS Thesis, AFIT/GE/ENG/90D-20. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1990.

10. Fairbanks, Ronald R., Capt, Headquarters 6585th Test Group. Holloman AFB, NM. Letter and Unclassified Empirical Radar Information. 19 June 1990.

11. Flynn, P. M. *Alternative Dynamics Models and Multiple Model Filtering for a Short Range Tracker.* MS Thesis, AFIT/GE/ENG/81D. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1981.

12. Gill, T. P. *The Doppler Effect.* New York: Academic Press, 1965.

13. Harnly, Douglas A., and Robert L. Jensen. *An Adaptive Distributed-Measurement Extended Kalman Filter for a Short Range Tracker.* MS Thesis, AFIT/GA/EE/79-1. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1979.

14. Herrera, Theodore D. *Kalman Filter Tracking of a Reflective Target Using Forward Looking Infrared Measurements and Doppler Returns.* MS Thesis, AFIT/GE/ENG/91D-25. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1991.

15. Integrated Systems Inc. *MATRIX$_X$ 7.0 Release Notes.* Integrated Systems Inc., Santa Clara, California, October 1988.

16. Kozemchak, Mark R. *Enhanced Image Tracking. Analysis of Two Acceleration Models in Tracking Multiple Hot-Spot Images.* MS Thesis, AFIT/GE/ENG/82D. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1982.

17. Leeney, T. A. *A Multiple Model Adaptive Tracking Algorithm Against Airborne Targets.* MS Thesis, AFIT/GE/ENG/87D-37. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1987.

18. Levanon, Nadav. *Radar Principles.* New York: John Wiley and Sons, 1988.

19. Loving, Phyllis A. *Bayesian vs. MAP Multiple Model Adaptive Estimation for Field of View Expansion in Tracking Airborne Targets.* MS Thesis, AFIT/GE/ENG/85M. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, March 1985.

20. McManamon, Paul F. *Active/Passive Sensor for Strategic Defense.* Electro-Optics Branch, Mission Avionics Division, Wright Laboratories, Wright-Patterson AFB, OH, 16 April 1991.

21. Maybeck, Peter S. *Stochastic Models, Estimation and Control, Volume I.* New York: Academic Press, 1979.

22. - - - - -. *Stochastic Models, Estimation and Control, Volume II.* New York: Academic Press, 1982.

23. - - - - -. "Adaptive Tracking of Maneuvering Targets Based on IR Image Data." *Kalman Filter Integration of Modern Guidance and Navigation Systems*, Proceedings of the North Atlantic Treaty Organization (NATO), AGARD Lecture Series No. 166, 7-1 - 7-18, London, England, 1989.

24. - - - - -. Professor of Electrical Engineering. Personal Interviews. Air Force Institute of Technology (AU), School of Engineering, Wright-Patterson AFB, OH, January through December 1992.

25. - - - - -, D. A. Harnly and R. L. Jensen. "An Adaptive Extended Kalman Filter for Target Image Tracking," *IEEE Transactions on Aerospace and Electronic Systems*, Vol AES-17, No. 2, 173-180, March 1981.

26. - - - - - and D. E. Mercier. "A Target Tracker Using Spatially Distributed Infrared Measurements," *IEEE Transactions on Automatic Control*, Vol. AC-25, No. 2, 222-225, April 1980.

27. - - - - - and S. K. Rogers. "Adaptive Tracking of Multiple Hot-Spot Target IR Images," *IEEE Transactions on Automatic Control*, Vol. AC-28, No. 10, 937-943, October 1983.

28. - - - - - and R. I. Suizu. "Adaptive Tracker Field of View Variation Via Multiple Model Filtering," *IEEE Transactions of Aerospace and Electronic Systems*, Vol. AES-21, No. 4, 529-539, July 1985.

29. - - - - -, W. H. Worsley, and P. M. Flynn. "Investigation of Constant Turn-Rate Dynamics Models in Filters for Airborne Vehicle Tracking," *Proceedings of the IEEE National Aerospace and Electronics Conference*, Dayton, OH, 896-903, May 1982.

30. - - - - - and others. AFIT Graduate Student Developed Software for Kalman Filter Tracking of Airborne Targets Using Infrared and Laser Sensors. Fortran Software, Air Force Institute of Technology (AU), School of Engineering, Wright-Patterson AFB, OH, 1978 - December 1992.

31. Mercier, D. E. *An Extended Kalman Filter for Use in a Shared Aperature Medium Range Tracker.* MS Thesis, AFIT/GA/EE/78D-3. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1978.

32. Miller, Judith. *Doppler Tracking Program.* Briefing Slides. Phillips Lab, Kirtland AFB, New Mexico, 27 February 1991.

33. Millner, P. P. *Enhanced Tracking of Airborne Targets Using a Correlator/Kalman Filter.* MS Thesis, AFIT/GE/ENG/82M. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1982.

34. Murphy, P. J. *Laser Scattering and Extinction Techniques for Rocket Plume Diagnostics.* Technical Report AEDC-TR-89-16. Arnold Engineering Development Center,Air Force Systems Command, Arnold Air Force Base, Tennesse, April 1990.

35. Musick, Stanton. Personal Interviews. Reference Systems Branch, Avionics Laboratory, Wright Laboratories, Wright-Patterson AFB, OH, April through December 1992.

36. Nawrocki, Paul J. and Robert Papa. *Atmospheric Processes.* New Jersey: Prentice-Hall Inc., 1963.

37. Netzer, Allan S. *Characteristics of Bayesian Multiple Model Adaptive Estimation for Tracking Airborne Targets.* MS Thesis, AFIT/GAE/ENG/85D-2. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1985.

38. Norton, John E. *Multiple Adaptive Tracking of Airborne Targets.* MS Thesis, AFIT/GA/ENG/88D-2. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1988.

39. Nowakowski, J. and A. Kalisz. *Measurement of Doppler Signatures.* Technical Report FR-1/B78-705. Air Force Weapons Laboratory (AFSC), Kirland AFB, New Mexico, 15 August 1990.

40. Rizzo, David R. *Enhanced Tracking of Ballistic Targets Using Forward Looking Infrared Measurements.* MS Thesis, AFIT/GE/ENG/89M-7. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, March 1989.

41. Rogers, Steven K. *Advanced Tracking of Airborne Targets Using Forward Looking Infrared Measurements.* MS Thesis, AFIT/GE/ENG/81D. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1981.

42. Singletery, J. *Adaptive Laser Pointing and Tracking Problem.* MS Thesis, AFIT/GEO/EE/ENG/80D-12. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1980.

43. Skolnik, Merril I. *Introduction to Radar Systems.* New York: McGraw-Hill, Inc., 1962.

44. Stimson, George W. *Introduction to Airborne Radar.* Hughes Aircraft Company, El Segundo, CA., 1983.

45. Suizu, Robert I. *Advanced Tracking of Airborne Targets Using Multiple Model Adaptive Filtering Techniques for Adaptive Field of View Expansion.* MS Thesis, AFIT/GE/ENG/83D. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH December 1983.

46. The MathWorks, Inc. *Pro-*MATLAB™ *for Sun Workstations.* The MathWorks Inc., South Natick, Massacheusetts, January 31, 1990.

47. Tobin, David M. *A Multiple Model Adaptive Tracking Algorithm for a High Energy Laser Weapon System.* MS Thesis, AFIT/GE/ENG/86D-37. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1986.

48. Tobin, David M., and Peter S. Maybeck. "Substantial Enhancements to a Multiple Model Adaptive Estimator for Tracking Image Tracking," *Proceedings of the IEEE Conference on Decision and Control*, Los Angeles, CA, 2002-2011, December 1987.

49. Tobin, David M., and Peter S. Maybeck. "Enhancements to a Multiple Model Adaptive Estimator/Image Tracker," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-24, No. 4, pp 417-426, July 1988.

50. Toomay, J. C. *Radar Principles for the Non-Specialist.* Belmont, Ca: Wadsworth, Inc., 1982.

51. Wlodawski, M., et. al. *Measurements of Scattering by Particulates in Rocket Exhaust Plumes.* Technical Report TR-1/B78-7-05, Air Force Weapons Laboratory, Kirtland AFB, New Mexico, 23 January 1990.

52. Wolfram Research Inc. *Mathematica™ A System for Doing Mathematics by Computer.* Addison-Wesley Publishing Company, Inc., Redwood City, California, 1988.

Vita

Michael L.G. Ching was born in Honolulu Hawaii on March 28, 1963. Despite being a U.S. Army brat, he got to stay in Hawaii for seven years before moving to the Far East for several years and then back to the states in 1974. He made a couple more stateside moves and ended up graduating from James Madison High School in San Antonio, Texas in 1981.

He received a lot of things other than a Bachelor of Sciences Degree in Electrical Engineering while attending the United States Air Force Academy. After being commissioned in the U.S. Air Force with the Class of 1985, he headed for his first assignment in the Mojave Desert. For three years, Capt Ching developed and tested instrumentation systems for the 6510th Test Wing, Edwards Air Force Base, California. He remained with the same unit for three more years while evaluating prototype avionics systems before entering the Master of Sciences graduate program at the Air Force Institute of Technology. Upon receiving his degree, Capt Ching will be assigned to the Phillips Laboratory, Kirtland Air Force Base, New Mexico.

Permanent Address: 4814 El Gusto
San Antonio, TX 78233

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
15 December 1992

3. REPORT TYPE AND DATES COVERED
Master's Thesis

4. TITLE AND SUBTITLE
Kalman Filter Tracking of A Ballistic Missile Using
Forward Looking Infrared Measurements and Doppler Returns

5. FUNDING NUMBERS

6. AUTHOR(S)
Michael L.G. Ching
Captain, USAF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology
WPAFB OH 45433-6583

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/GE/ENG/92D-09

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Capt Joseph Bergin
Phillips Laboratory/LITT Kirtland AFB, NM 87117

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The Air Force Institute of Technology (AFIT) has been involved in developing Kalman Filter trackers for airborne targets for the last 14 years. The goal of this particular thesis was to track a ballistic missile in the boost phase at ranges up to 2000 km, in order to control a high energy laser weapon designed to destroy the target. The filter developed combined an existing "FLIR" filter, which estimated location of the plume intensity centroid based on measurements from a forward looking infrared sensor (FLIR), and an existing "center-of-mass" filter, which estimated the offset between the plume and missile center-of-mass based on measurements from low-energy laser reflections. In addition, the new filter modeled the oscillation of the rocket plume with respect to the missile hardbody, known as the "pogo" affect, in the hopes of improving overall tracking performance. Filter performance is analyzed through use of Monte Carlo simulation software developed at AFIT. This thesis also performed observability tests on various filter configurations in order to gain insight into observability problems identified during earlier research. Observability of states is measured through the use of both stochastic observability testing and Monte Carlo analysis of the filter models using the Multimode Simulation for Optimal Filter Evaluation (MSOFE) software.

14. SUBJECT TERMS
Kalman Filter, Tracking, Infrared, Doppler, Laser

15. NUMBER OF PAGES
289

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

BEST AVAILABLE COPY

GENERAL INSTRUCTIONS FOR COMPLETING SF 298

The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistent with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to ***stay within the lines*** to meet *optical scanning requirements*.

Block 1. Agency Use Only *(Leave blank)*.

Block 2. Report Date. Full publication date including day, month, and year, if available (e.g. 1 Jan 88). Must cite at least the year.

Block 3. Type of Report and Dates Covered. State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).

Block 4. Title and Subtitle. A title is taken from the part of the report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

Block 5. Funding Numbers. To include contract and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:

C	Contract	PR	Project
G	Grant	TA	Task
PE	Program Element	WU	Work Unit Accession No.

Block 6. Author(s). Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler, this should follow the name(s).

Block 7. Performing Organization Name(s) and Address(es). Self explanatory.

Block 8. Performing Organization Report Number. Enter the unique alphanumeric report number(s) assigned by the organization performing the report.

Block 9. Sponsoring/Monitoring Agency Name(s) and Address(es). Self-explanatory.

Block 10. Sponsoring/Monitoring Agency Report Number. *(If known)*

Block 11. Supplementary Notes. Enter information not included elsewhere such as: Prepared in cooperation with...; Trans. of...; To be published in.... When a report is revised, include a statement whether the new report supersedes or supplements the older report.

Block 12a. Distribution/Availability Statement. Denotes public availability or limitations. Cite any availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR).

DOD -	See DoDD 5230.24, "Distribution Statements on Technical Documents."
DOE -	See authorities.
NASA -	See Handbook NHB 2200.2.
NTIS -	Leave blank.

Block 12b. Distribution Code.

DOD -	Leave blank.
DOE -	Enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical Reports.
NASA -	Leave blank.
NTIS -	Leave blank.

Block 13. Abstract. Include a brief *(Maximum 200 words)* factual summary of the most significant information contained in the report.

Block 14. Subject Terms. Keywords or phrases identifying major subjects in the report.

Block 15. Number of Pages. Enter the total number of pages.

Block 16. Price Code. Enter appropriate price code *(NTIS only)*.

Blocks 17. - 19. Security Classifications. Self-explanatory. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED). If form contains classified information, stamp classification on the top and bottom of the page.

Block 20. Limitation of Abstract. This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be unlimited.

BEST AVAILABLE COPY